



Introduction

This user manual describes the demonstration firmware running on the STM32373C-EVAL evaluation board, which can be used to evaluate the capabilities of the STM32F373VC(T6) microcontroller and on-board peripherals.

This demo contains many applications that can be easily reused, such as electrocardiogram acquisition, accurate temperature and pressure measurement, dual interface (I2C and RF) EEPROM application (ESL and DataLogger), HDMI-CEC, infrared RC5 and SIRC receiver and transmitter, RTC calendar, file system FAT implementation on SD Card, wave player using STM32 I2S peripheral, voice recording using the Sigma Delta Analog to Digital Converter, temperature sensor interfacing and TFT LCD.

The STM32373C-EVAL demonstration firmware is provided with the demonstration programmed in the internal Flash memory, and all the files needed by the demonstration are programmed in the MicroSD card. The demonstration is executed at each reset (board power-up, external reset, etc.).

In case the STM32373C-EVAL board was not factory-programmed, or the demonstration application was erased, or the STM32373C-EVAL board configuration needs to be modified, the Bootloader, IAP or STM32 STLink Utility can be used to program this file. For more details, refer to [Section 3: STM32373C-EVAL demonstration package](#) and [Section 4: STM32373C-EVAL demonstration programming](#).

Note: Before you execute the demonstration, make sure that all EVAL board jumpers are well configured. For more details, refer to [Section 1.10.11: STM32373C-EVAL board jumper configuration](#).

This demonstration firmware and other firmware packages are available for download from the STMicroelectronics website: www.st.com.

Table 1. Applicable tools

Type	Part number
Evaluation tools	STM32373C-EVAL

Contents

- 1 Functional description 7**
 - 1.1 Power control 8
 - 1.2 Clocking 8
 - 1.3 Reset control 8
 - 1.4 Debug JTAG interface 8
 - 1.5 Serial wire debugger interface 8
 - 1.6 Embedded ST-LINK 8
 - 1.7 Display devices 8
 - 1.7.1 LCD 8
 - 1.7.2 LEDs 8
 - 1.7.3 LDR (Light Dependent Resistor) 9
 - 1.8 Interfaces 9
 - 1.8.1 RS232 9
 - 1.9 IrDA 9
 - 1.10 Miscellaneous peripherals 9
 - 1.10.1 Joystick 9
 - 1.10.2 Push-buttons 9
 - 1.10.3 Potentiometer 9
 - 1.10.4 Audio 9
 - 1.10.5 MicroSD card 9
 - 1.10.6 RF EEPROM 9
 - 1.10.7 IR LED & IR receiver 10
 - 1.10.8 HDMI CEC 10
 - 1.10.9 Temperature sensor 10
 - 1.10.10 Touch slider 10
 - 1.10.11 STM32373C-EVAL board jumper configuration 10

- 2 Running the demonstration 11**
 - 2.1 Menu 11
 - 2.1.1 Demo startup 13
 - 2.1.2 Navigation 16
 - 2.2 Clock sources 17
 - 2.2.1 Clock control 17

2.2.2	Clock failure	18
2.3	STM32F373VC(T6) resources	19
2.3.1	Peripherals	19
2.3.2	Interrupts	20
2.3.3	External interrupts	21
2.3.4	Internal memory size	21
2.3.5	External memory organization	21
2.4	Demo applications	24
2.4.1	Organizer	24
2.4.2	Image Viewer submenu	31
2.4.3	Audio	32
2.4.4	Connectivity submenu	36
2.4.5	Thermometer	42
2.4.6	Low-power modes	44
2.4.7	RF EEPROM	51
2.4.8	Brightness application	57
2.4.9	Sigma-Delta applications	58
2.4.10	Help	62
2.4.11	About submenu	64
3	STM32373C-EVAL demonstration package	65
4	STM32373C-EVAL demonstration programming	67
4.1	Programming the media files	67
4.2	Programming the demo	67
4.2.1	Using Bootloader	67
4.2.2	Using IAP	67
4.2.3	Using preconfigured projects	68
5	Revision history	69

List of tables

Table 1.	Applicable tools.	1
Table 2.	STM32F373VCT6) demo peripherals.	19
Table 4.	STM32F373VC(T6) demo external interrupts.	21
Table 5.	M24LR64 memory organization (ESL application)	55
Table 6.	Document revision history	69

List of figures

Figure 1.	Evaluation board overview	7
Figure 2.	Structure of the demonstration menus	12
Figure 3.	SD card check	13
Figure 4.	Warning message	13
Figure 5.	ST logo	14
Figure 6.	STM32F3 presentation slide	14
Figure 7.	Time and date configuration	14
Figure 8.	Main menu	15
Figure 9.	Corresponding submenus	15
Figure 10.	Navigating in the demonstration menus	16
Figure 11.	Clock tree diagram	17
Figure 12.	No HSE clock detected	18
Figure 13.	Standby mode entered	18
Figure 14.	Internal Flash memory organization	21
Figure 15.	MicroSD card organization	23
Figure 16.	SD card removal	23
Figure 17.	Organizer menu	24
Figure 18.	Setting the time and date	24
Figure 19.	Time Adjust submenu	25
Figure 20.	Time Show submenu	26
Figure 21.	Setting the year	27
Figure 22.	Setting the month	27
Figure 23.	Setting the day of the month	27
Figure 24.	Exiting the Date Show submenu	28
Figure 25.	StopWatch sub menu	28
Figure 26.	Lap timer submenu	29
Figure 27.	Setting the alarm activation time	30
Figure 28.	Alarm Show submenu	30
Figure 29.	Message displayed if time and date need setting	31
Figure 30.	Image Viewer submenu	31
Figure 31.	STM32 Image Viewer	32
Figure 32.	Wave player submenu	33
Figure 33.	Wave player interface	33
Figure 34.	Wave player playing submenu	34
Figure 35.	Pause submenu	34
Figure 36.	Voice recording submenu selected	35
Figure 37.	Record submenu	35
Figure 38.	Starting wave record	36
Figure 39.	Player submenu	36
Figure 40.	HDMI CEC submenu selected	37
Figure 41.	HDMI CEC configuration submenu	37
Figure 42.	CEC menu	37
Figure 43.	Select CEC command	38
Figure 44.	Receive subscreen information	38
Figure 45.	IR transmitter menu	39
Figure 46.	IR transmitter command menu	40
Figure 47.	IR receiver menu	40
Figure 48.	IR receiver application menu	41

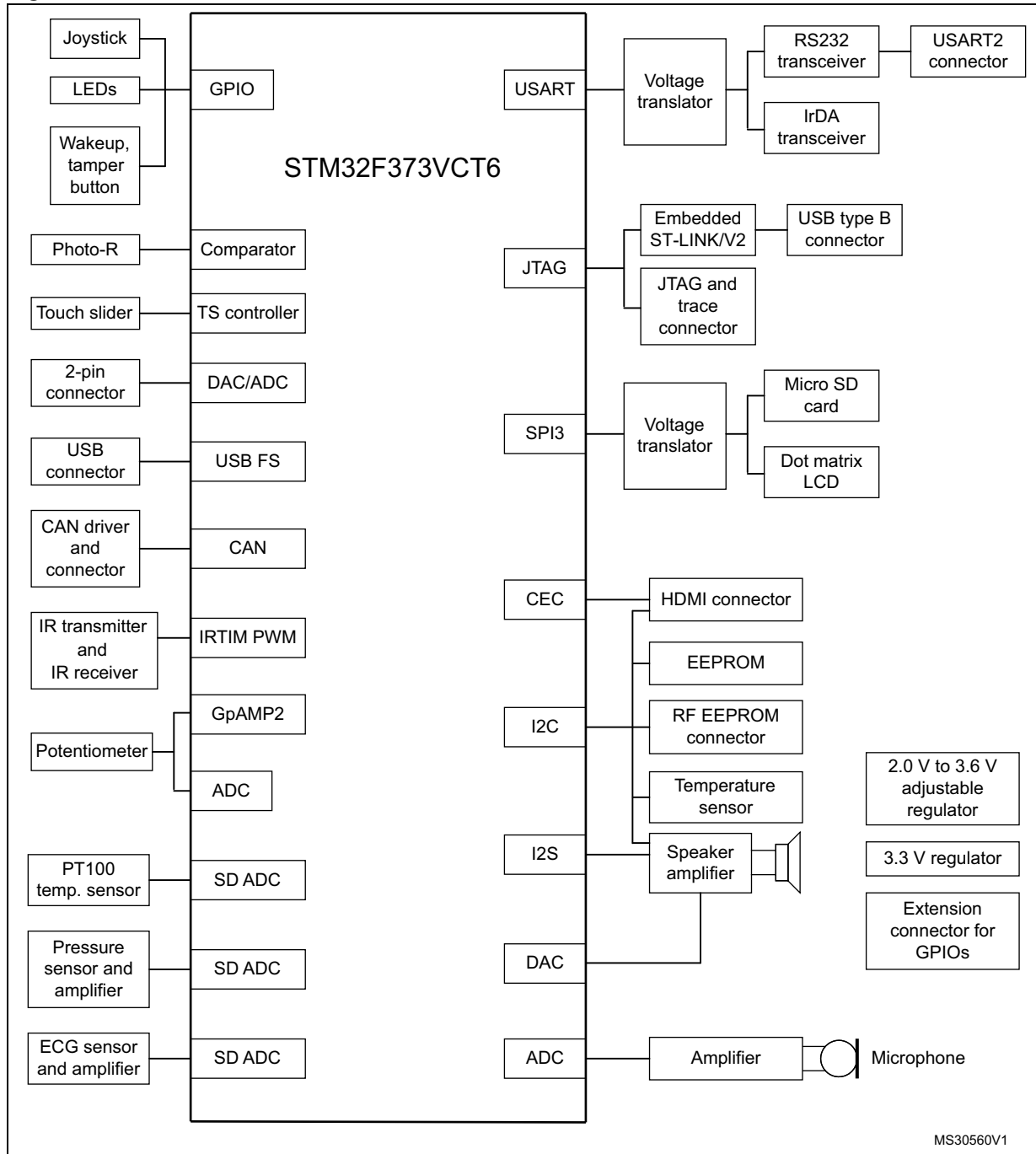
Figure 49.	IR receiver command menu	41
Figure 50.	USB mass storage menu 1	42
Figure 51.	USB mass storage menu 2	42
Figure 52.	Thermometer submenu selected	43
Figure 53.	Temperature display	43
Figure 54.	Warning temperature display	44
Figure 55.	Temperature sensor error	44
Figure 56.	Exiting Stop mode	45
Figure 57.	Stop mode entered exit EXTI	45
Figure 58.	MCU in the Stop mode Exit EXTI	46
Figure 59.	RTC Alarm causes the MCU to exit Stop mode	46
Figure 60.	Setting the wakeup time	47
Figure 61.	RTC Alarm wakeup configured	47
Figure 62.	RTC Alarm wakeup	48
Figure 63.	Time and date configuration prompt	48
Figure 64.	Entering Standby mode	49
Figure 65.	MCU in Standby mode	49
Figure 66.	RTC Alarm causes the MCU to exit the Standby mode	50
Figure 67.	Setting the wakeup time	50
Figure 68.	RTC Alarm wakeup configured	51
Figure 69.	Time and date configuration prompt	51
Figure 70.	RF EEPROM daughter board	52
Figure 71.	M24LR64-R block diagram	52
Figure 72.	RF EEPROM menu	53
Figure 73.	ESL application	53
Figure 74.	ESL setting menu	54
Figure 75.	Communication block diagram	55
Figure 76.	DataLogger sub-menu	56
Figure 77.	DataLogger dialog box	56
Figure 78.	DataLogger curve	57
Figure 79.	Brightness level	58
Figure 80.	Sigma-Delta sub-menu	58
Figure 81.	ECG waveform display	59
Figure 82.	Temperature calibration phase	60
Figure 83.	Temperature measurement phase	60
Figure 84.	Temperature display	61
Figure 85.	Pressure display	61
Figure 86.	Help menu	62
Figure 87.	Navigation menu 1	62
Figure 88.	Navigation menu 2	63
Figure 89.	Jumper configuration menu 1	63
Figure 90.	Jumper configuration menu 2	64
Figure 91.	About submenu	64
Figure 92.	STM373C-EVAL demo package directory tree	65
Figure 93.	SD Card directory organization	67

1 Functional description

The STM32373C-EVAL evaluation board provides a development and demonstration platform for STM32F373VC-based applications. It is designed to allow the user to try out the major functions of the STM32F373VC(T6) microcontroller.

Figure 1 summarizes the main functional blocks of the evaluation board.

Figure 1. Evaluation board overview



1.1 Power control

The evaluation board can be powered from an external 5 V supply, from the USB connector or ST-Link connector. All other required voltages are provided by on-board voltage regulators.

1.2 Clocking

Two clock sources are available on the STM32373C-EVAL evaluation board:

- 32 KHz crystal for embedded RTC
- 8 MHz crystal for the STM32F373VC main clock system

1.3 Reset control

The reset can be generated by hardware or software:

- Reset button: activates the RESET input when pressed
- JTAG reset

1.4 Debug JTAG interface

Software debug is done via the standard ARM[®] JTAG connection: 20-pin IDC (insulation displacement connector) for connection to the standard ARM host interface.

1.5 Serial wire debugger interface

The Serial Wire Debug Port (SWD-DP) provides a 2-pin (clock + data) interface to the AHP-AP port.

1.6 Embedded ST-LINK

An embedded ST-LINK is integrated on the board as an embedded in-circuit debugger and programmer for the STM32F373VC(T6) MCU.

1.7 Display devices

1.7.1 LCD

A TFT color LCD module is mounted on the STM32373C-EVAL board. It is interfaced through the embedded SPI peripheral.

1.7.2 LEDs

Four general-purpose LEDs are available. They are used as a display.

1.7.3 LDR (Light Dependent Resistor)

The VDDA supply is divided by a resistor bridge consisting of an LDR VT9ON1 and an 8.2K resistor and is connected to PA1.

1.8 Interfaces

1.8.1 RS232

The STM32F373 evaluation board (STM32373C-EVAL) provides one on-board RS-232 serial port. RS232 port (USART2) is accessed via the CN12 connector.

1.9 IrDA

The STM32373C-EVAL evaluation board supports IrDA communication. The interface is mounted on USART2 (U12 interface).

1.10 Miscellaneous peripherals

1.10.1 Joystick

Four-direction joystick with a selection key.

1.10.2 Push-buttons

The following push-buttons are available:

- Key
- Tamper

1.10.3 Potentiometer

Varistor: A Successive Approximation ADC channel (ADC1_IN9)/Sigma Delta ADC channel (SDADC1_5P) is connected to an on-board variable resistor. The variable resistor provides a voltage in the range of 0 V to 3.3 V.

1.10.4 Audio

The STM32373C-EVAL evaluation board supports stereo audio play, using an audio DAC CS43L22 connected to both I2S2 ports, and one channel of the DAC embedded in the STM32F373VC microcontroller.

1.10.5 MicroSD card

The STM32373C-EVAL evaluation board has a MicroSD card connector connected to the SPI peripheral.

1.10.6 RF EEPROM

RF EEPROM daughter board implemented on the module is the M24LR64-R. The daughter

board can be connected on CN3 to the STM32F373VC(T6) via the I2C bus.

The I2C address of the RF EEPROM is 0b1010E2E1E0. The E0-E2 values are determined by the RF EEPROM daughter board.

1.10.7 IR LED & IR receiver

The IR receiver TSOP34836 is connected to PB5 on the STM32F373VC(T6) and a current of around 100mA on the IR LED is driven by PB9 through transistors T6 and T7 on the board.

1.10.8 HDMI CEC

Two HDMI connectors CN1 (sink) and CN2 (source) are available on the STM32373C-EVAL board.

1.10.9 Temperature sensor

The STM32373C-EVAL evaluation board includes an I²C temperature sensor connected to the I2C2 peripheral.

1.10.10 Touch slider

The STM32373C-EVAL evaluation board provides support for a touch slider connected to 4 capacitive sensing channels.

1.10.11 STM32373C-EVAL board jumper configuration

To be able to run the STM32373C-EVAL demo correctly, configure the following STM32373C-EVAL board jumpers as follows:

- JP1: fitted
- JP4: fitted in position 1<->2
- JP5: fitted in position 1<->2
- JP11: fitted in position 1<->2
- JP12: fitted in position 2<->3
- JP13: fitted in position 2<->3
- JP15: fitted
- JP16: fitted
- JP17: fitted

Note: Set JP11 in BAT position to save time even if the evaluation board is powered off.

2 Running the demonstration

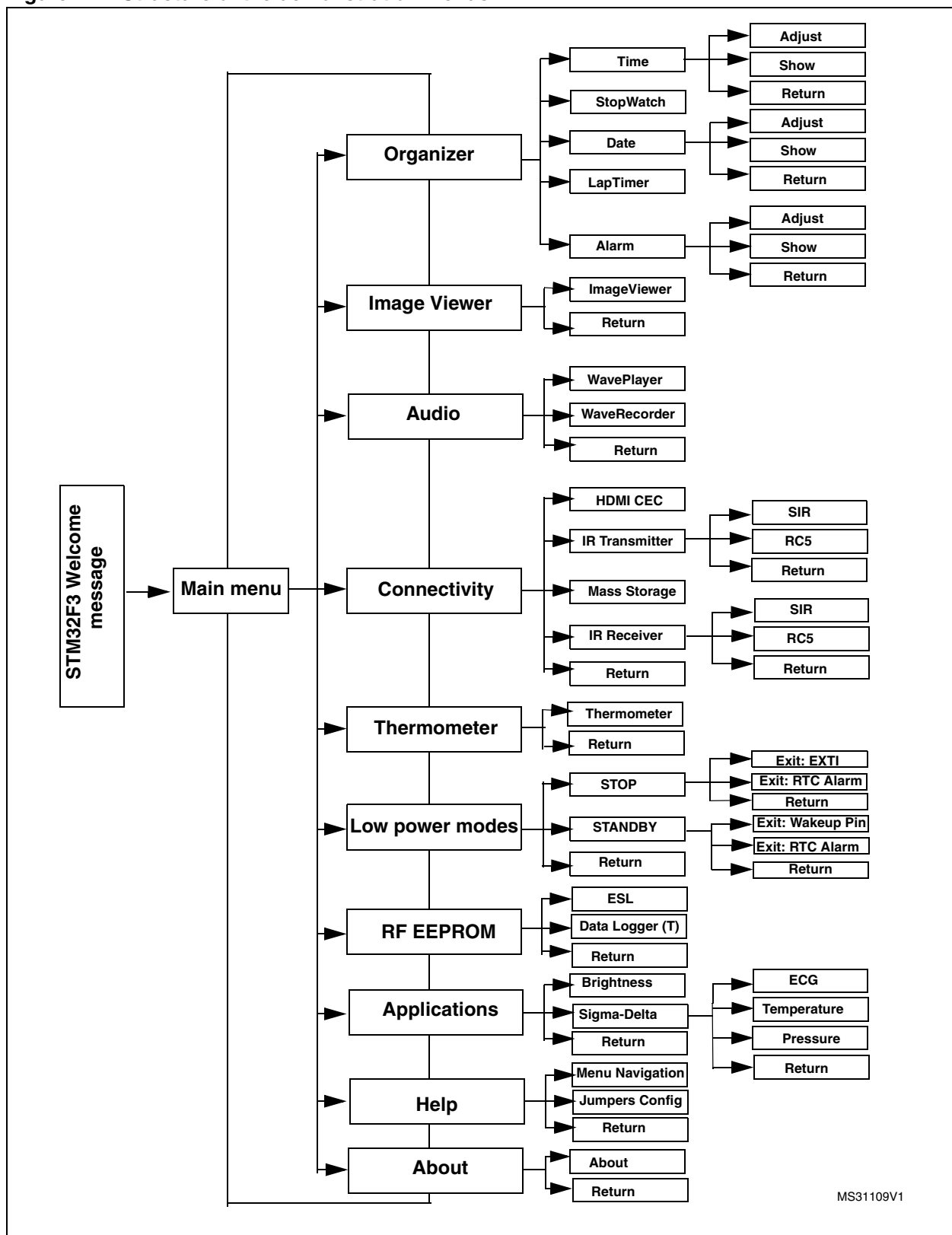
2.1 Menu

Figure 2 shows the menu system of the STM32373C demonstration. The main menu is shown on the left-hand side. The UP, DOWN, RIGHT and LEFT joystick directions allow the user to navigate between items in the main menu and the submenus. To enter a submenu, press the SEL push-button.

The action of pressing the SEL push-button is performed by vertically pressing the top of the joystick, as opposed to moving it horizontally UP, DOWN, RIGHT or LEFT.

To exit a submenu, select the Return menu and press SEL.

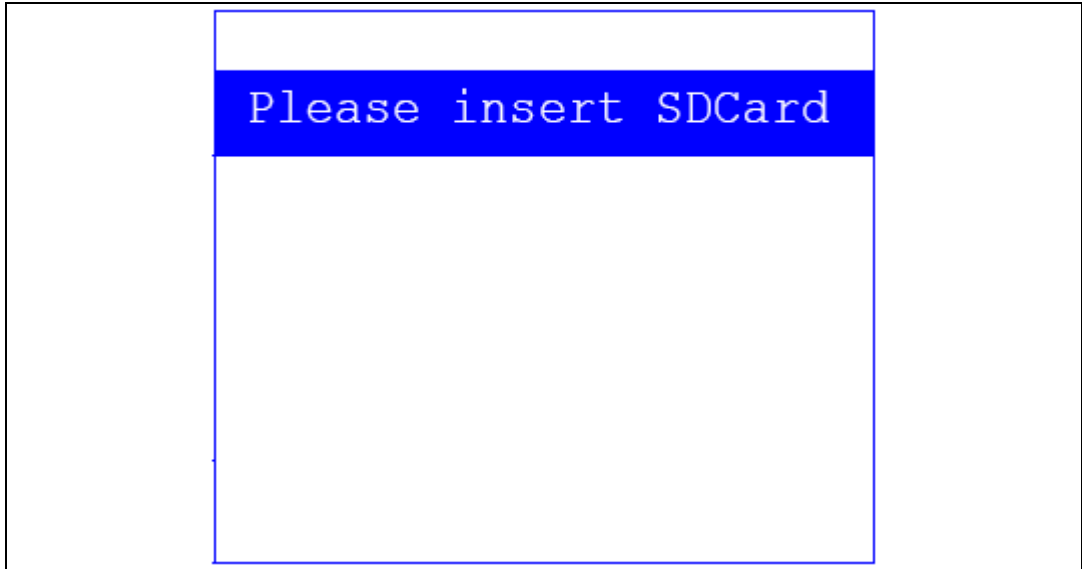
Figure 2. Structure of the demonstration menus



2.1.1 Demo startup

After a board reset, at demo startup, the system checks if an SD memory card is already present in connector CN7. If no card is detected, the demo does not start and the message shown in [Figure 3](#) is displayed on the LCD screen.

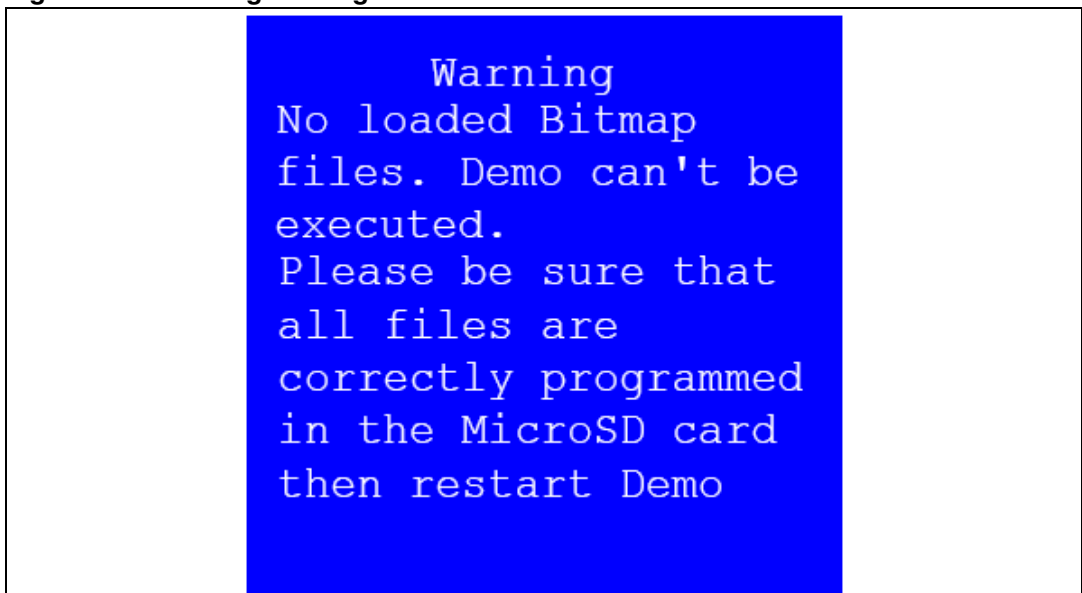
Figure 3. SD card check



The demo continues only if an SD card is inserted.

Then, the demo graphic icons and bitmap files are checked in the MicroSD card (see [Section 2.3.5: External memory organization](#)). All the icons have to be correctly programmed in the MicroSD card for the demo to start, so if an icon is missing, the demo does not start and the message shown in [Figure 4](#) is displayed on the LCD screen.

Figure 4. Warning message



However, if the icons are correctly loaded in the SD Card memory, the welcome screen is displayed and the ST logo appears on the LCD (see [Figure 5](#)).

Figure 5. ST logo



Then, after one second, an STM32F3 presentation slide is displayed on the LCD screen.

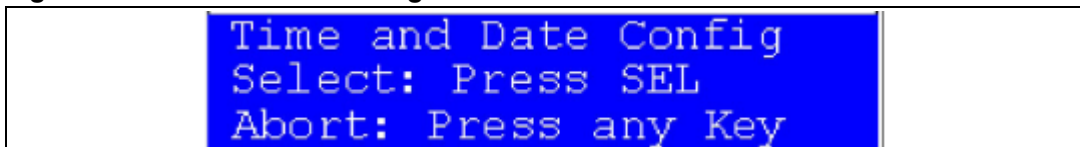
Figure 6. STM32F3 presentation slide



When the board is powered up for the first time, the user is prompted to set the time, year, month and day. The user may choose to ignore it by pressing any key except for the SEL push-button to abort the configuration sequence. To set the time and date, the user must press SEL and follow the setting sequence.

The message shown in [Figure 7](#) appears on the LCD screen.

Figure 7. Time and date configuration



Note: 1 If the user chooses to configure the time and date, the Time Adjust and Date Adjust menus are displayed. Otherwise, the main menu is displayed and the user can set the time

parameters in the organizer menu. To set the time/date, use the joystick UP/DOWN and SEL push-buttons.

- 2 If the time configuration has already been done, then the number of elapsed days (higher than 1 day) from the last time the demo board was powered up appears on the LCD screen. It is soon followed by the current date.

Once the time/date have been set, the main menu appears. The main menu is displayed in the form of a set of icons. It shows all the submenus in the same screen. You can navigate using the UP, DOWN, RIGHT and LEFT joystick directions to select the required submenu. To enter a submenu, press the SEL joystick push-button, and the new submenu corresponding to the selected icon is displayed.

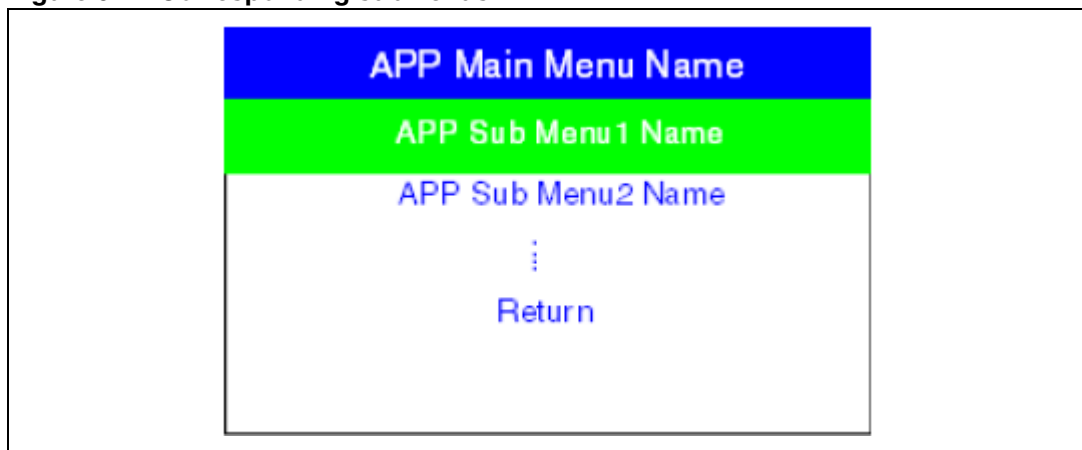
Figure 8. Main menu



Note: The icons shown in Figure 8 in are taken from http://commons.wikimedia.org/wiki/Crystal_Clear.

Once a submenu has been selected, the name of the application is listed at the top of the display and all the corresponding submenus are listed below as shown in Figure 9.

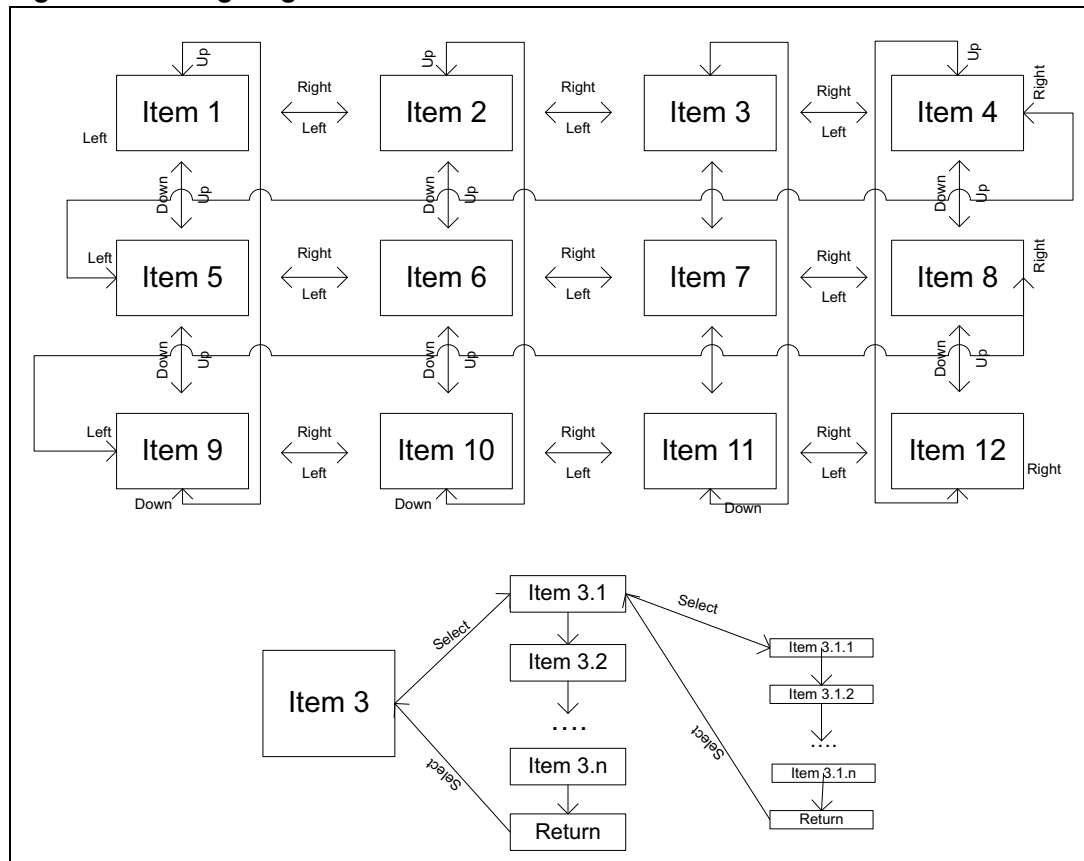
Figure 9. Corresponding submenus



2.1.2 Navigation

The demonstration menu is based on circular navigation, submenu selection, item selection and back navigation as described in *Figure 10*.

Figure 10. Navigating in the demonstration menus



The user navigates using the joystick push-buttons located on the evaluation board: RIGHT, LEFT, SEL, UP and DOWN.

- The UP, DOWN, RIGHT and LEFT push-buttons are used to perform circular navigation in the main menu and the current menu items.
- The SEL push-button selects the current item.
- The UP and DOWN push-buttons are used for vertical navigation in the submenus.
- To return to the upper menu, go to the Return line and press SEL.

2.2 Clock sources

2.2.1 Clock control

The STM32F373VC(T6) internal clocks are derived from the HSE (clocked by the external 8 MHz crystal).

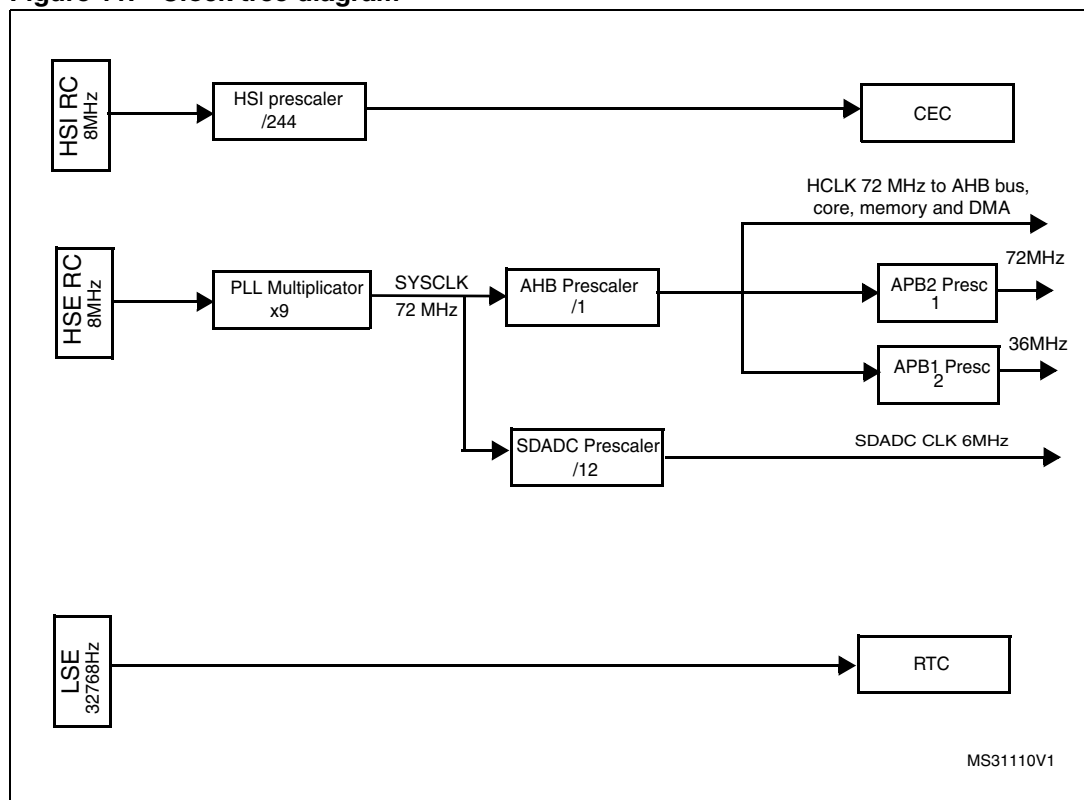
In this demo application, the various system clocks are configured as follows:

- System clock is set to 72 MHz: the PLL is used as the system clock source: 72 MHz.
- HCLK frequency is set to 72 MHz.
- Timer clock (TIMCLK) is set to 72 MHz.
- SDADC clock is set to 6 MHz.
- CEC clock is set to 32 KHz (HSI/244).
- PCLK1 is set to 36 MHz.
- PCLK2 is set to 72 MHz.

Only the RTC is clocked by a 32 kHz external oscillator.

Figure 11 illustrates the clock tree organization for this demo.

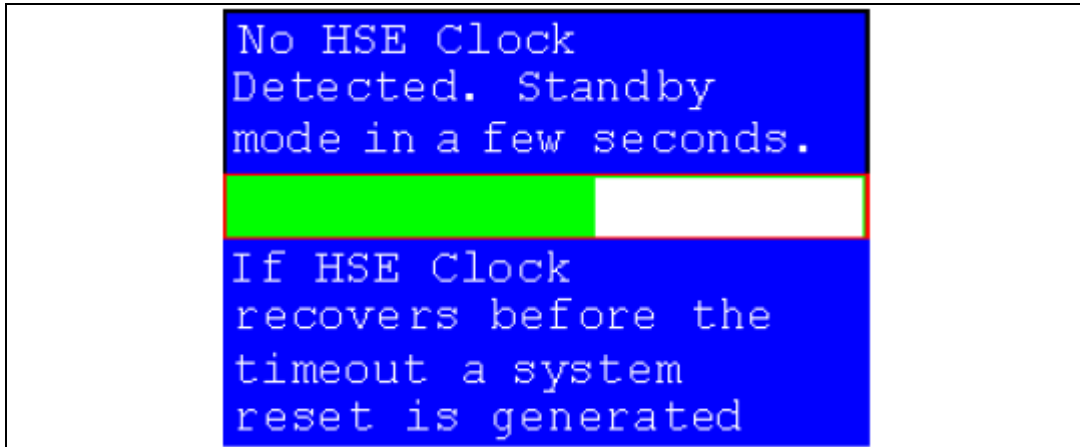
Figure 11. Clock tree diagram



2.2.2 Clock failure

At any demo level, if no clock is present on OSC_IN (broken or disconnected crystal), the message shown in *Figure 12* is displayed on the LCD screen.

Figure 12. No HSE clock detected

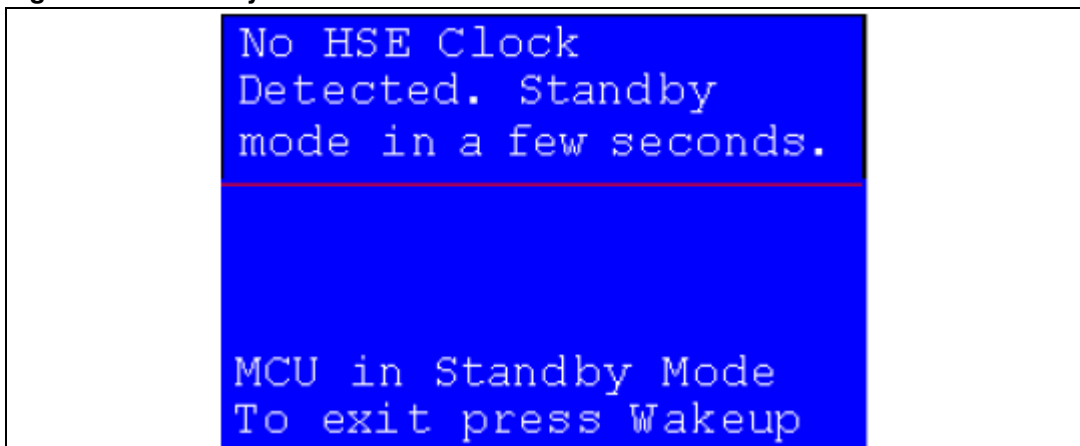


If the 8 MHz crystal is not reconnected in the next few seconds, the MCU enters Standby mode. If the 8 MHz crystal is reconnected within a few seconds, a system reset is generated.

Note: The clock security system (CSS) feeds the MCU with the HSI OSC used as an emergency clock if no clock is detected.

When a timeout occurs, the MCU enters Standby mode and the message shown in *Figure 13* is displayed on the LCD screen.

Figure 13. Standby mode entered



Note: The demo does not restart as long as the 8 MHz crystal is not present. Connecting the 8 MHz crystal oscillator after reset may not restart the demo correctly. The crystal oscillator must be connected before starting the demo.

2.3 STM32F373VC(T6) resources

2.3.1 Peripherals

All used peripherals are described in [Table 1](#).

Table 2. STM32F373VCT6) demo peripherals

Used peripherals	Application
CEC	CEC demo
COMP	Brightness level detection (brightness level)
DAC1	Brightness level detection (defines the comparator level)
DAC2	Electrocardiogram
DMA1	Wave player
DMA2	Electrocardiogram
EXTI	Menu navigation + joystick + push-button + low power mode+ wave player + applications
GPIO	All applications + LEDs
I2C2	Temperature sensor (STLM75), dual interface EEPROM and CEC
I2S	Wave player
NVIC	All applications using interrupts
PWR	Low power modes
RCC	All applications + Demo kernel
RTC	Organizer (calendar, stop-watch, lap timer, alarm)
SDADC1	Voice recorder, Temperature measurement (PT100), pressure measurement and electrocardiogram
SPI3	MicroSD + Color LCD
SysTick	Generate 10 ms time base
TIM2	LED toggling
TIM3	Electrocardiogram, infra-red receiver and CEC demo
TIM4 and TIM13	Voice recording
TIM16 and TIM17	Infra-red transmitter
TIM19	Pressure measurement
USB	Mass Storage

2.3.2 Interrupts

Table 2 shows all the enabled interrupts.

Table 3. STM32F373VC(T6) demo interrupts

Interrupts	Priority	Used for
CEC	Priority: 0 Sub priority: 1	CEC interrupt
DMA1	Priority: 0 Sub priority: 0	Wave player
DMA2	Priority: 0 Sub priority: 0	Electrocardiogram
EXTI0		Wakeup button
EXTI2_TS	Priority: 2 Sub priority: 0	Menu navigation
EXTI3	Priority: 0 Sub priority: 0	SD Card detection
EXTI4	Priority: 2 Sub priority: 0	Menu navigation
EXTI9_5	Priority: 3 Sub priority: 0	Menu navigation
EXTI15_10	Priority: 2 Sub priority: 0	Menu navigation
I2C2 Error	Priority: 0 Sub priority: 0	SMBus Alert interrupt
NMI	Priority: -2	CSS interrupt
RTC	Priority: 0 Sub priority: 0	Calendar, date update, and Alarm generation
SDADC1	Priority: 0 Sub priority: 0	Pressure measurement
SysTick	Priority: 0 Sub priority: 0	System timing
Tamper	Priority: 0 Sub priority: 0	Tamper generation
TIM2	Priority: 3 Sub priority: 3	LED toggling interrupt
TIM3	Priority: 0 Sub priority: 0	Infra-red receiver
TIM4	Priority: 1 Sub priority: 1	Wave recorder (write in file) interrupt
TIM13	Priority: 0 Sub priority: 0	Wave recorder (write in buffer) interrupt

Table 3. STM32F373VC(T6) demo interrupts (continued)

Interrupts	Priority	Used for
TIM16	Priority: 0 Sub priority: 0	Infra-red Transmitter
USB	Priority: 0 Sub priority: 0	Mass storage

2.3.3 External interrupts

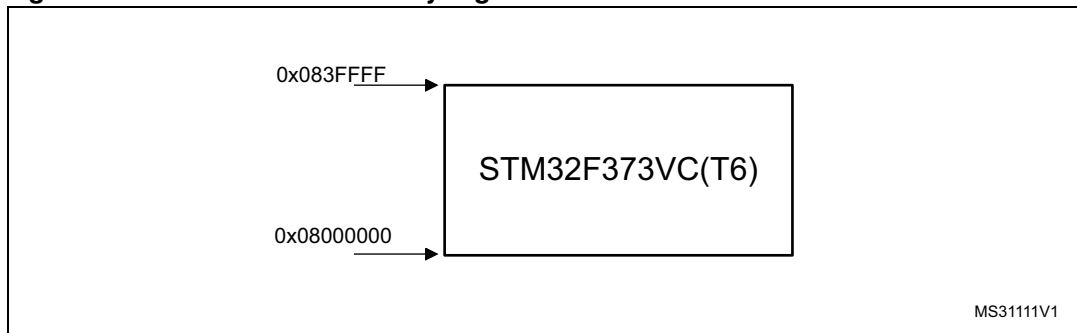
Table 3 shows all the external interrupts used by the demonstration.

Table 4. STM32F373VC(T6) demo external interrupts

External interrupts	Used for
EXTI line 0	Tamper (interrupt mode, falling edge)
EXTI line 2	Key Button (interrupt mode, rising edge) Joystick DOWN (interrupt mode, falling edge)
EXTI line 3	SD Card detect (interrupt mode, rising edge)
EXTI line 9	Joystick SEL ((interrupt mode, falling edge)
EXTI line 10	Joystick UP (interrupt mode, falling edge)
EXTI line 17	RTC Alarm (interrupt mode, rising edge)

2.3.4 Internal memory size

Figure 14. Internal Flash memory organization



2.3.5 External memory organization

The STM32373C-EVAL demo is based on an embedded free FAT file system, FatFs. The file system is needed to read all media information from the on-board MicroSD card memory. The SD card memory is organized in three subdirectories:

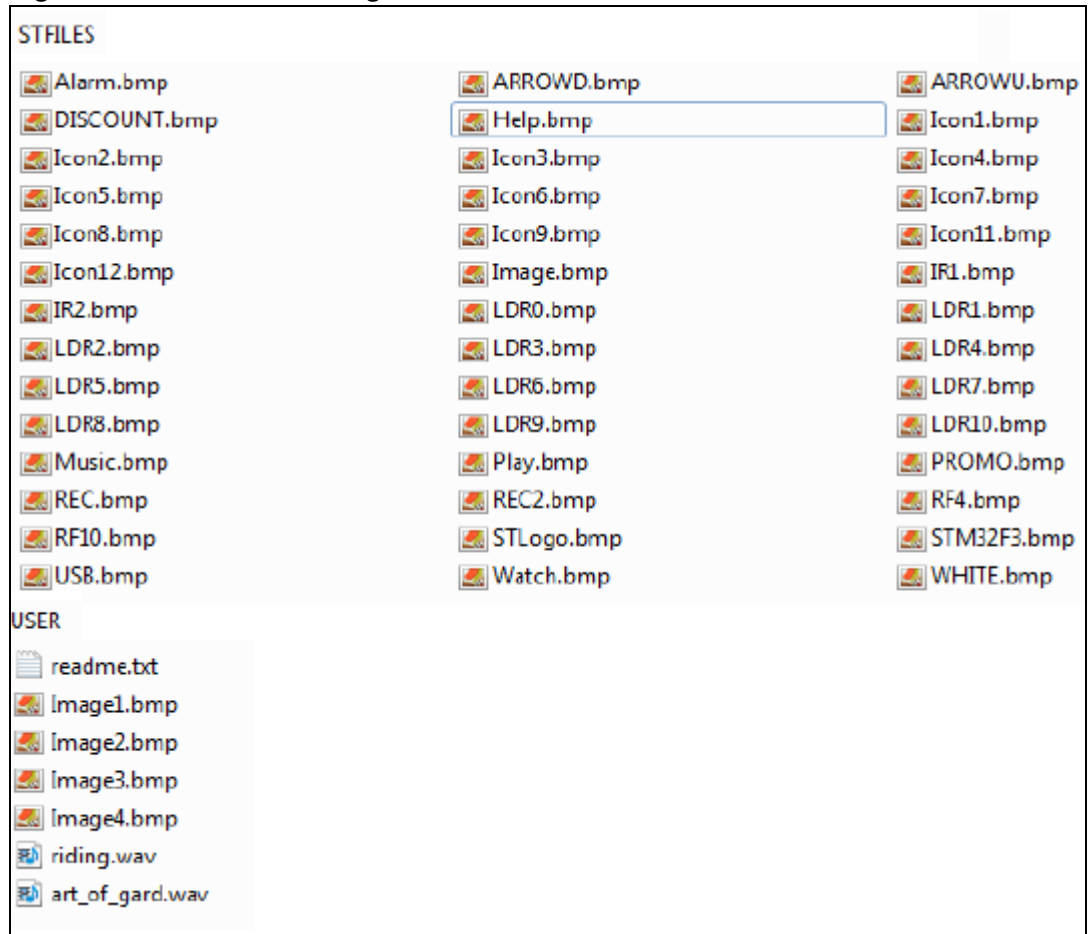
- STFILES: this directory contains all the required demo media files (icons). User files located in this folder cannot be handled by the demo; only default files are managed.
- USER: this is a user folder. The user can add his/her own files here to be played inside the demo menus (pictures and waves). This folder is used only by the Image Viewer

and Wave Player submenus. For more details on the various files properties, please refer to [Section 2.4.2: Image Viewer submenu](#) and [Section 2.4.3: Audio](#).

- REC: this directory contains the voice recorded wave file "Rec_wave.wav".(This Folder is not present by default, it is created when the Voice Recording application is run).

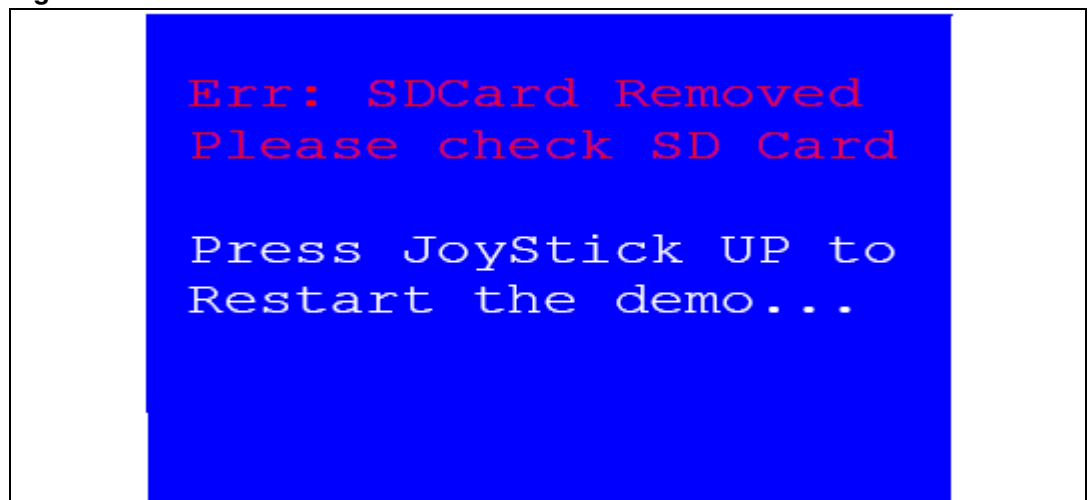
Note: The STFILES directory and its internal files are mandatory for demo startup. FatFs is a generic FAT file system module for small embedded systems. The FatFs is written in compliance with ANSI C and completely separated from the disk I/O layer. For more details, refer to the following link: http://elm-chan.org/fsw/ff/00index_e.html.

Figure 15. MicroSD card organization



Note: The user can add his/her 16-bit bitmap images (320x240) and wave files in the USER folder. At any demo level, if the SD card is removed, the demo stops and the message shown in Figure 16 is displayed on the LCD screen.

Figure 16. SD card removal



2.4 Demo applications

The following section provides a detailed description of each part of the demonstration.

Note: In the demonstration, the core runs at HCLK = 72MHz.

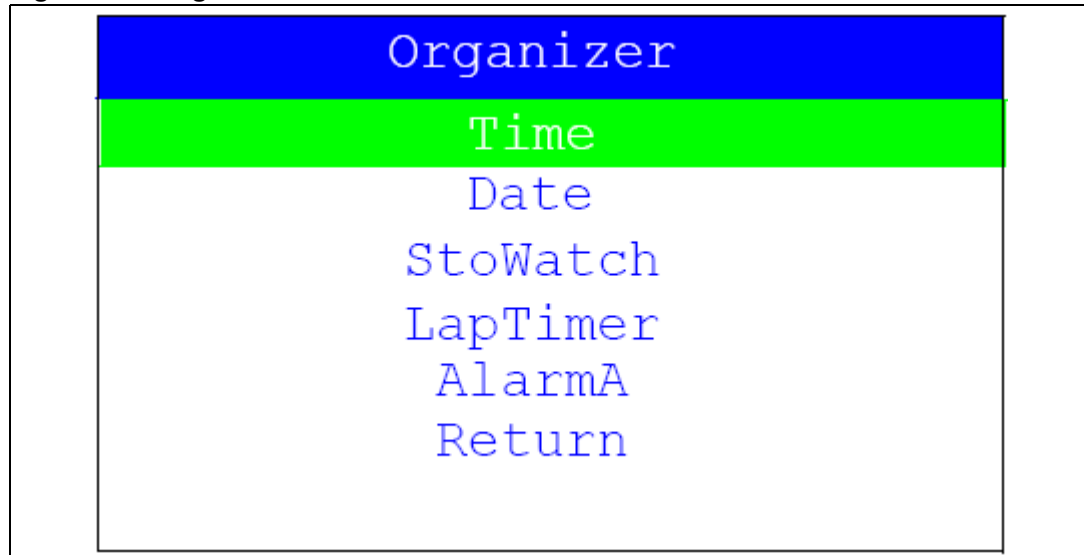
Four LEDs, LD1, LD2, LD3 and LD4, flash throughout the demonstration at a frequency depending on the core clock.

2.4.1 Organizer

The STM32F373VC(T6) features a real-time clock (RTC) which is an independent BCD timer/counter. The RTC provides a time-of-day clock/calendar, two programmable alarm interrupts, and a periodic programmable wakeup flag with interrupt capability.

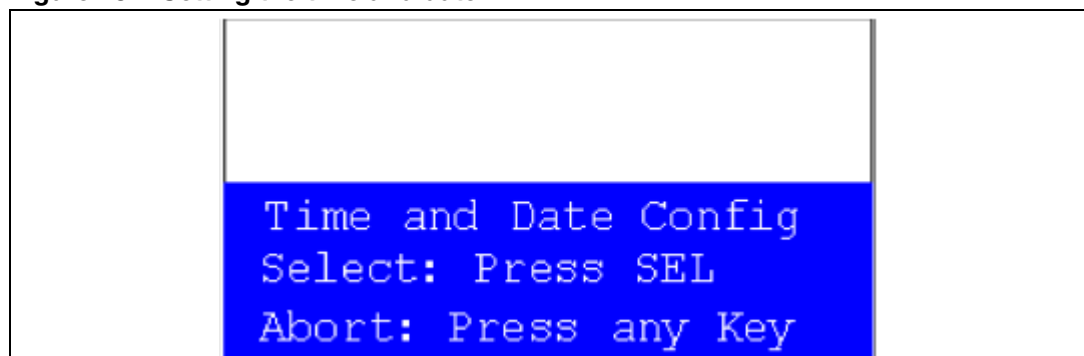
This submenu is used to configure/show the time and date, run stop-watch and lap timer subdemo and generate alarm. *Figure 17* shows the organizer menu.

Figure 17. Organizer menu



In any submenu, if the time and date parameters have not yet been configured, the message shown in *Figure 18* is displayed on the LCD screen.

Figure 18. Setting the time and date



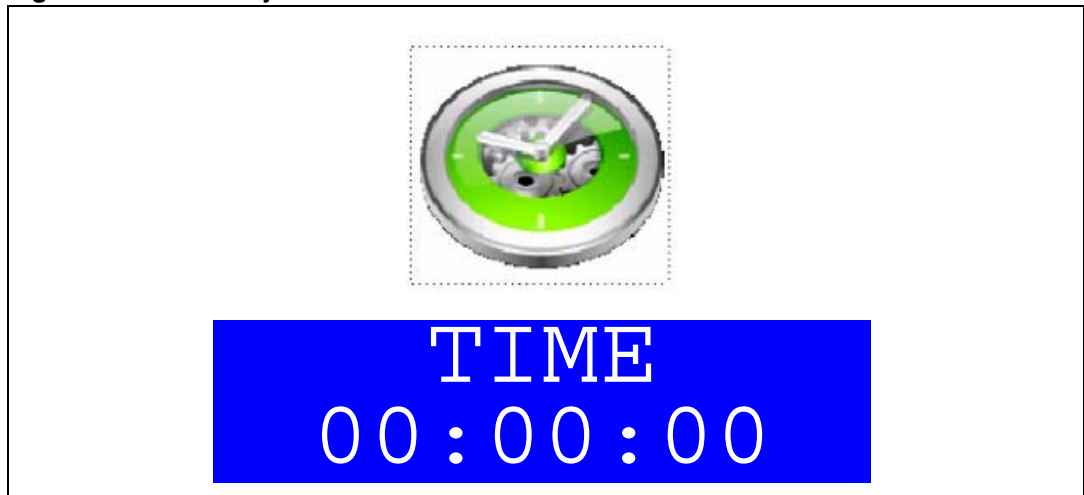
The user can optionally choose to set the time, year, month and day. Press any key (except for SEL) to ignore the prompt and abort the configuration sequence. Press on SEL and follow the setting sequence to set the time and date.

Time submenu

This submenu is divided into two items that allow the user to display or set the current time.

- **Time Adjust:** after the evaluation board is powered up, select this submenu to change the default time (00:00:00) to the current time. Once Time Adjust has been selected, the first digit of the hour field can be changed. Press the UP button to display the current value plus one. Press the DOWN button to display the previous digit value. After setting the digit value, press SEL, and the cursor automatically jumps to the next digit. When all the time digits have been set, the Time submenu appears. Some digit values are limited to a range of values depending on the field (hour, minutes or seconds). The following message (with the default time or the current time) is displayed on the LCD when this submenu is selected.
- Set JP11 in BAT position to save time even if the evaluation board is powered off

Figure 19. Time Adjust submenu



- **Time Show:** this item displays the current time. If time and date have not been configured before, a message is displayed, prompting the user to set the time and date or to exit to the upper submenu. When this submenu is selected, the message shown in [Figure 20](#) appears on the LCD. In the example, the time has not been set yet.

Figure 20. Time Show submenu

To exit the Time Show submenu, press the SEL push-button. To exit the Time submenu, select the Return line and press the SEL push-button.

Date submenu

This submenu is divided into two items that allow the user to display or set the current date.

- **Date Adjust:** select this item after each power-up in order to set the current date. If the time and date have not been configured before, a message is displayed, prompting the user to set the time and date or to exit to the upper submenu.

The user is requested to set the current date to be stored in the application memory. The date is displayed as Year, Month, Week Nbr, Day Nbr (number of the day in the year) with the selected day shown in the month. There is no default date since the user has to set the date at least once.

Once the submenu has been selected, the user starts by setting the year, then the month and the day of the selected month. The month and the year are selected using the UP or DOWN push-button. For the day, the UP, DOWN, RIGHT and LEFT push-buttons can be used. Press the UP push-button to display the current value plus one; press the DOWN push-button to display the previous value. To confirm the selected month, press the SEL push-button. The display then jumps to the year configuration. The same procedure is applicable for the year configuration.

After configuring the day, press the SEL push-button to store the entered value and exit to the Date submenu. The current date value is then shown and you can change the setting if required. The messages shown in [Figure 21](#), [Figure 22](#) and [Figure 23](#) are successively displayed on the LCD when this submenu is selected.

Figure 21. Setting the year

Month		Year				
Week Nbr		Day Nbr				
Mo	Tu	We	Th	Fr	Sa	Su
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

UP/DOWN: Set Year

Figure 22. Setting the month

Month		Year				
Week Nbr		Day Nbr				
Mo	Tu	We	Th	Fr	Sa	Su
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

UP/DOWN: Set Month

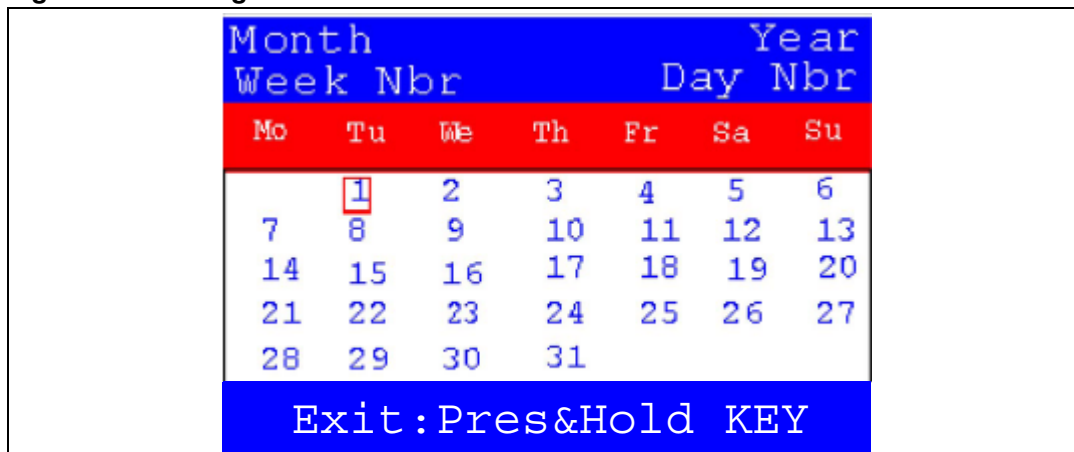
Figure 23. Setting the day of the month

Month		Year				
Week Nbr		Day Nbr				
Mo	Tu	We	Th	Fr	Sa	Su
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

Set Day-SEL to exit

- **Date Show:** this item displays the current date. If the time and date have not been configured before, a message is displayed, prompting the user to set the time and date or to exit to the upper submenu. The message shown in *Figure 24* is displayed on the LCD when the submenu is selected (with the date already configured).

Figure 24. Exiting the Date Show submenu

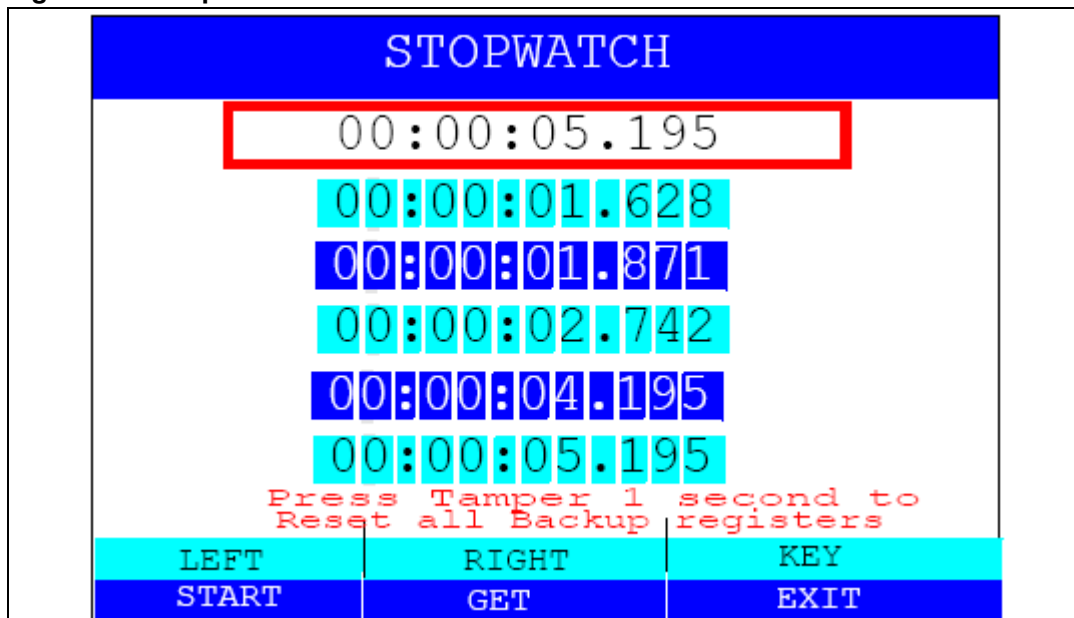


To exit this submenu, press the SEL push-button. To exit the Date submenu, select the Return line and press the SEL push-button.

Stopwatch submenu

This application simulates a precise chronometer with provision for 5 record times. For this application an interactive human interface is developed using STM32373C-EVAL LCD and push-buttons to allow user to use the stop-watch with real time display.

Figure 25. StopWatch sub menu



After startup, a default 00:00:00:000 chronometer counter is displayed on the LCD, it correspond to [Hours]:[minutes]:[seconds]:[milliseconds].

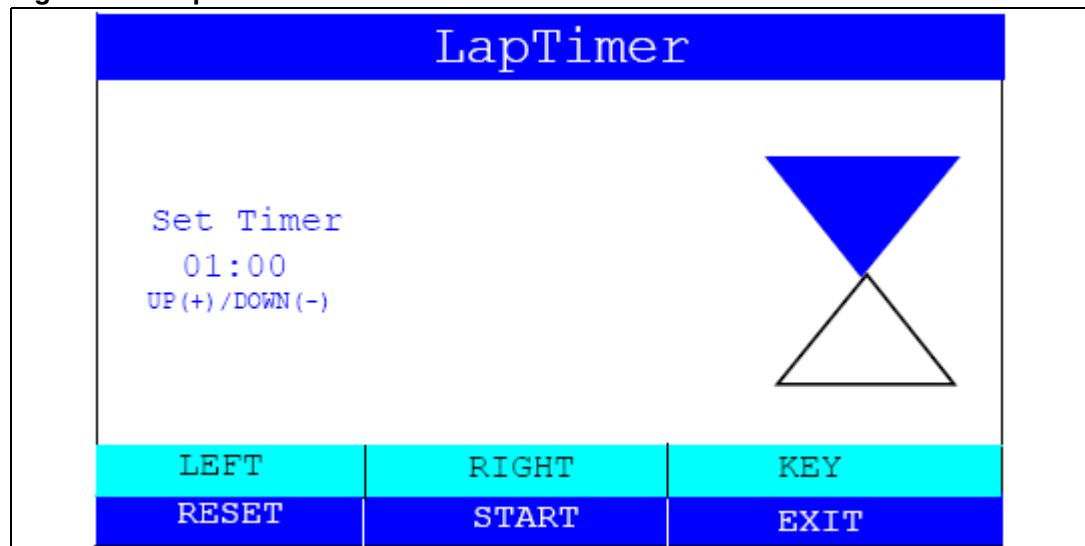
The user can control the chronometer features using the joystick LEFT and RIGHT push buttons:

- Press the joystick LEFT button to start the counter.
- Press the joystick RIGHT button to save trials in the backup registers (max 5 actions).
- Press the KEY button to exit.
- Press and hold the Tamper button for 1 sec to reset all the backup registers.

Lap timer

This application simulates a precise Hourglass, it measure the passage of a short period (sub seconds, seconds or minutes) of time. For this application an interactive human interface is developed using the STM32373C-EVAL LCD and push buttons to allow the user to use the lap timer with real time display. After startup, the Hourglass is filled and the default timer duration is set to one minute. Using the UP and DOWN push-buttons, the user can change the timer duration (the timer duration must be greater than 30 seconds and less than 3 minutes).

Figure 26. Lap timer submenu



The user can control the Hourglass features using the Joystick LEFT, RIGHT, UP and DOWN buttons:

- Press the joystick UP/DOWN button to adjust the timer (the UP and DOWN buttons are used only when the timer is reset).
- Press the joystick RIGHT button to start the timer.
- Press the joystick RIGHT button again to pause the timer.
- Press the joystick LEFT button to reset the timer.

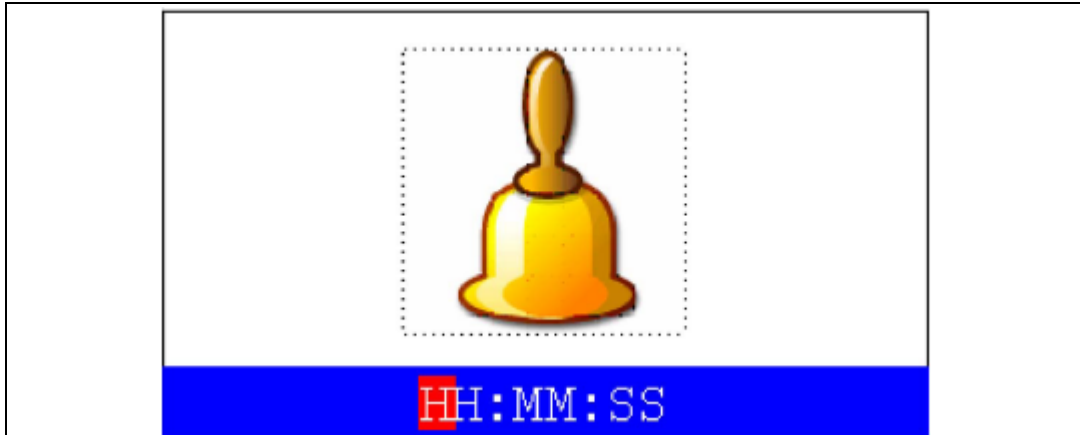
Alarm submenu

Using this submenu, the user can configure the alarm activation time. When the alarm time value is reached, all the LEDs (LED1 to LED4) start flashing together, and continue for 30 seconds. This submenu is divided into two items that allow the user to display or set the current alarm.

- **Alarm Adjust:** the alarm time activation is set in the same way as the time is set in the Time Adjust submenu. The following messages are successively displayed on the LCD

when the submenu shown in *Figure 27* is selected. When adjusting the alarm, the current time is displayed.

Figure 27. Setting the alarm activation time



- **Alarm Show:** this item displays the current alarm time. The default Alarm activation time displayed after power-up and is 00:00:00 before it has been set by the user in the Alarm Adjust submenu. If the time and date have not been configured before, a message shown in *Figure 29* is displayed. Pressing SEL takes you back to the Alarm submenu. The message shown in *Figure 28* is displayed on the LCD when this submenu is selected.

Figure 28. Alarm Show submenu



To exit the Alarm Show submenu, press the SEL push-button. To exit the Alarm submenu, select to the Return line and press the SEL push-button.

Note: In the Alarm Adjust and Alarm Show menus, if the time and date have not yet been configured, the message shown in *Figure 29* is displayed on the LCD screen.

Figure 29. Message displayed if time and date need setting

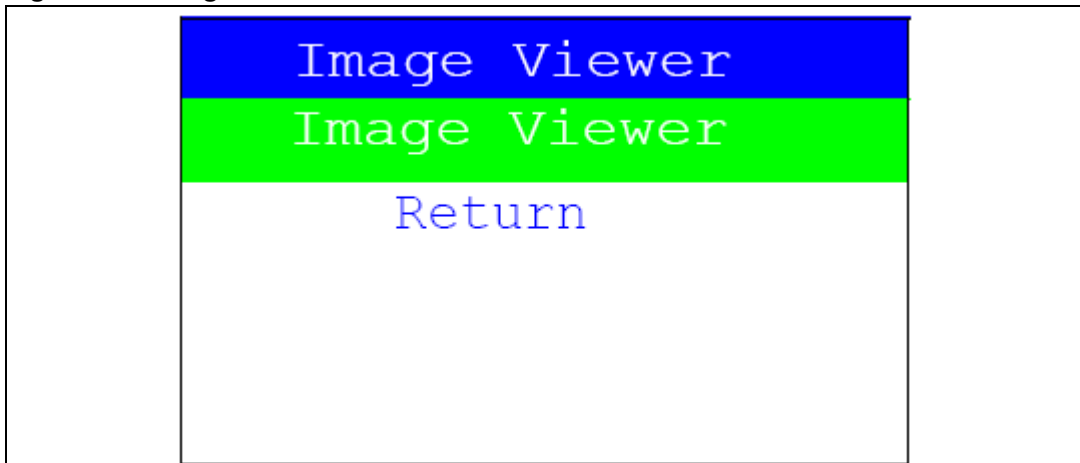
2.4.2 Image Viewer submenu

The Image Viewer submenu is used to demonstrate the LCD control performance using the embedded SPI interface. The application is a successive display of stored images.

This application reads all bitmap pictures from the USER directory (see [Section 4.1: Programming the media files](#)) and displays only the .BMP files having the following format:

- Bit depth: 16 bits (RGB)
- Size: 240x320

Select Image Viewer to display the submenu shown in [Figure 30](#).

Figure 30. Image Viewer submenu

When Image Viewer is selected, a list box of images is displayed as shown in [Figure 31](#).

Using the UP, DOWN and SEL push button the user can select and view any image from the list box.

Figure 31. STM32 Image Viewer

When Image Viewer is selected, the corresponding image is displayed and then the user can use RIGHT and LEFT push buttons to go to the next/previous image stored in the USER folder on the MicroSD card. If the DOWN push-button is pressed, the Image Viewer is stopped and the submenu shown in [Figure 30](#) is displayed.

The supported image size is 240x320. The defined number of images that are read from the MicroSD card is 4 images, selected in alphabetic order.

2.4.3 Audio

Wave player submenu

The STM32373C-EVAL evaluation board supports stereo audio play using the audio Codec CS43L22 connected to I2S1 port and controlled by I2C1 interface.

In this demo, an audio wave file stored under the USER folder in the microSD card is opened using the FATFS file system and transferred to the internal SRAM block by block (512 bytes) using the SPI interface. The voice sampling period is read from the wave file header. An audio amplifier is connected to the DAC interface to play the stored wave files.

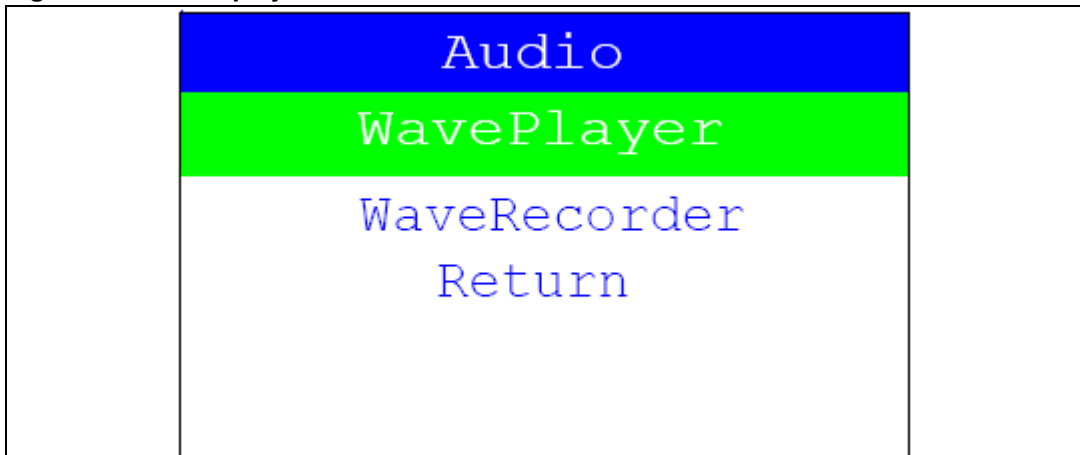
This application reads all wave files from “USER” directory (See [Section 4.1: Programming the Media files](#)) and only displays WAV files with the following format:

- Audio Format: PCM (an uncompressed wave data format in which each value represents the amplitude of the signal at the time of sampling.)
- Sample rate: **may be 8000, 11025, 22050 or 44100 Hz.**
- Bits Per Sample: 16-bit (Audio sample data values in the range [0-65535]).
- Number of channels: 2(Stereo)

The maximum number of wave files that can be read from the microSD card is 25 files selected in alphabetic order.

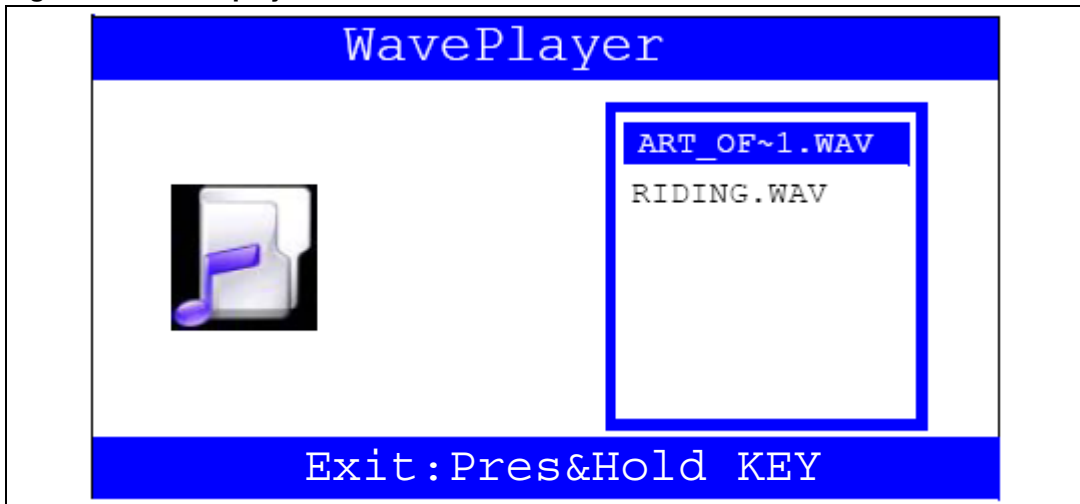
After selecting Wave Player, the submenu shown in [Figure 32](#) is displayed.

Figure 32. Wave player submenu



When Wave Player is selected again, a message box is displayed asking you to plug a headphone in connector CN21. Once done, the wave player file names are displayed in a list box as shown in *Figure 33*

Figure 33. Wave player interface



Using the UP, DOWN and SEL push-buttons, you can select the wave file to be played.

Once the play command is prompted (SEL push-button), the submenu shown in *Figure 34*. is displayed.

Figure 34. Wave player playing submenu



At this application level, pressing:

- The SEL push-button pauses the audio stream
- The LEFT push-button decrements the audio stream
- The RIGHT push-button increments the audio stream
- The DOWN push-button exits the wave player submenu

When the audio stream is paused, the menu in [Figure 35](#) is displayed.

Figure 35. Pause submenu



To resume playing, press the SEL push-button. The menu shown in [Figure 34](#) is then displayed.

When the audio stream is stopped, the stream position is reset and the menu shown in [Figure 33](#) is displayed.

Note: The audio files provided with this package are based on a free music download from the www.DanoSongs.com website.

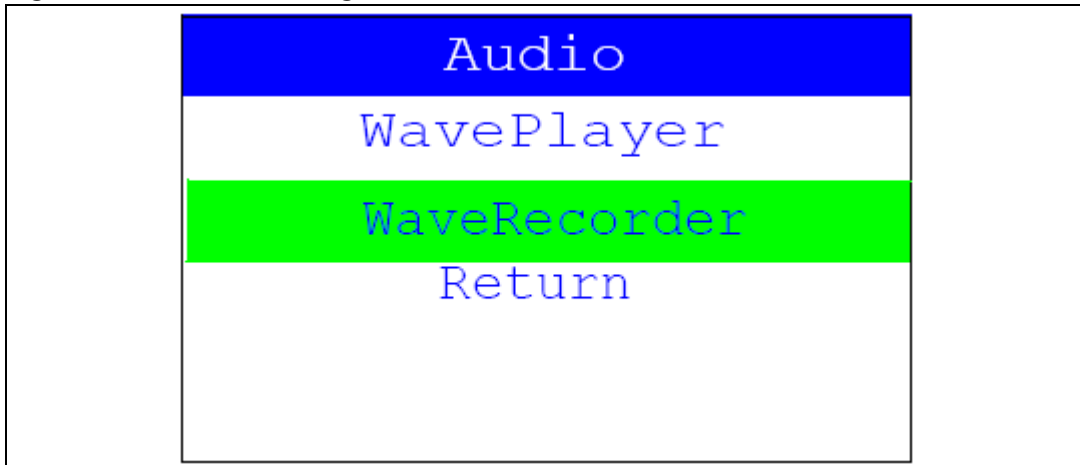
Wave recorder

The STM32F373VC(T6) microcontroller has an embedded 16-bit SDADC which can be used to record a voice (signal coming from the U28 microphone).

In this demo, the recorded wave buffers can be transferred from the internal SRAM block by block (512 bytes) and stored in the REC folder in the MicroSD card using the FatFs file system and using DMA and SPI. Timer 13 (TIM13) triggers the SDADC to convert the analog signal coming from the microphone each 8 KHz.

When the Wave Record submenu is selected, the message shown in [Figure 36](#) is displayed on the LCD.

Figure 36. Voice recording submenu selected



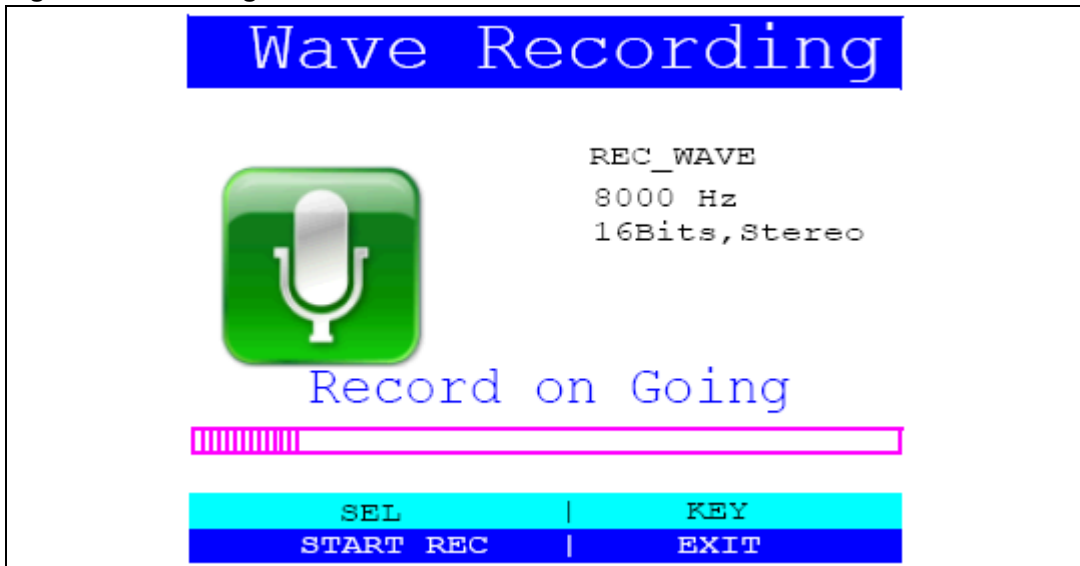
If you select the Record submenu by pressing the SEL push-button, the Voice Recording interface is displayed as shown in [Figure 37](#).

Figure 37. Record submenu



Once the record command is activated (by pressing on KEY push button), the submenu shown in [Figure 38](#) is displayed.

Figure 38. Starting wave record

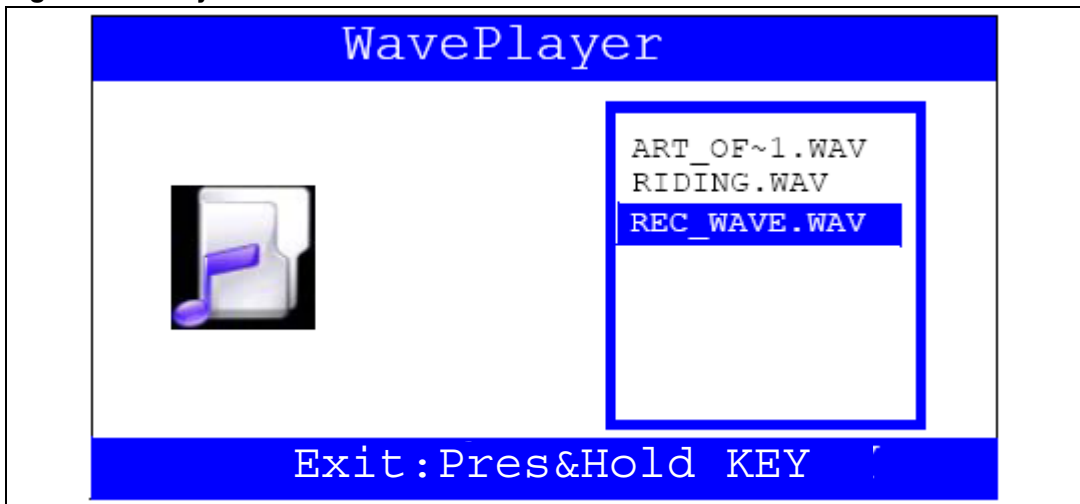


The recorded wave file properties are displayed on the right side of the Wave Record submenu.

When the KEY push-button is pressed, the menu in [Figure 36](#) is displayed.

The recorded wave is saved in an SD Card in the REC folder with name «REC_WAVE.WAV». To play the recorded wave, select the wave player sub-demo. The menu shown in [Figure 39](#) is displayed.

Figure 39. Player submenu



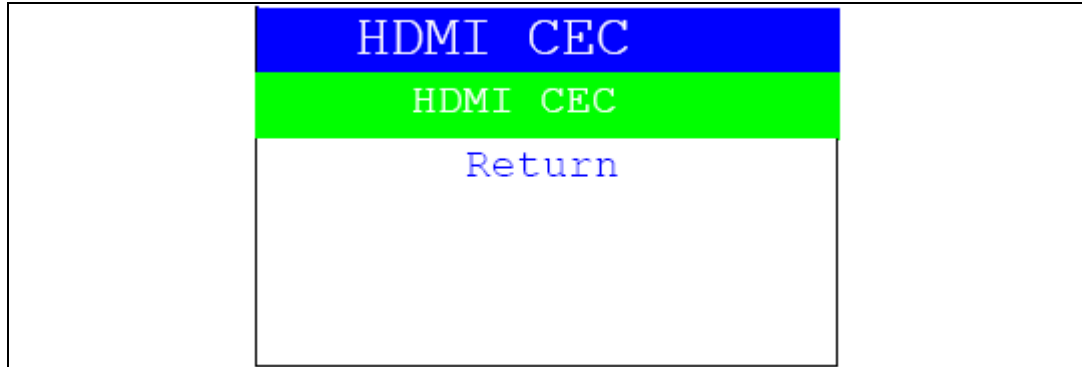
2.4.4 Connectivity submenu

HDMI™ CEC

The STM32F373VC microcontroller features an HDMI-CEC peripheral, this demonstration shows how to configure this peripheral and how to create CEC network providing a high level communication between different devices using CEC protocol messages.

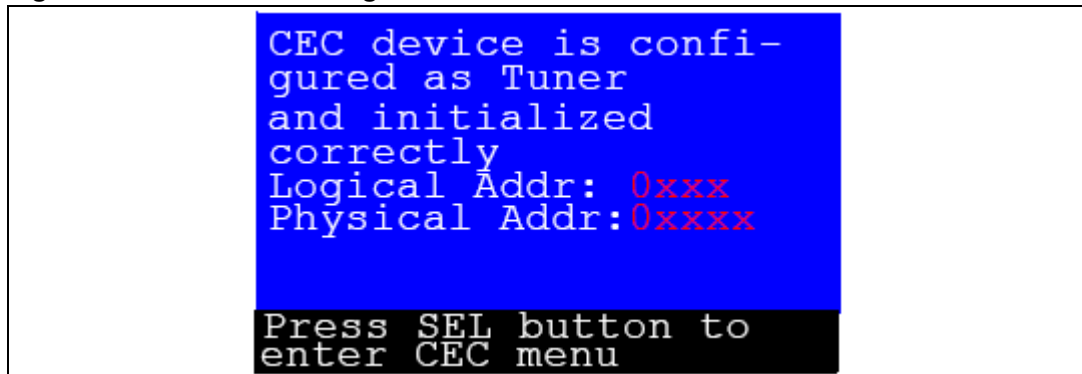
When the **HDMI CEC** submenu is selected, the message shown in *Figure 40* is displayed on the LCD.

Figure 40. HDMI CEC submenu selected



Once you select the **HDMI CEC** submenu, if no CEC error is generated, the device is configured as Tuner and the physical and logical addresses are displayed on the LCD as shown in *Figure 41*. To enter the CEC menu, press the SEL push-button.

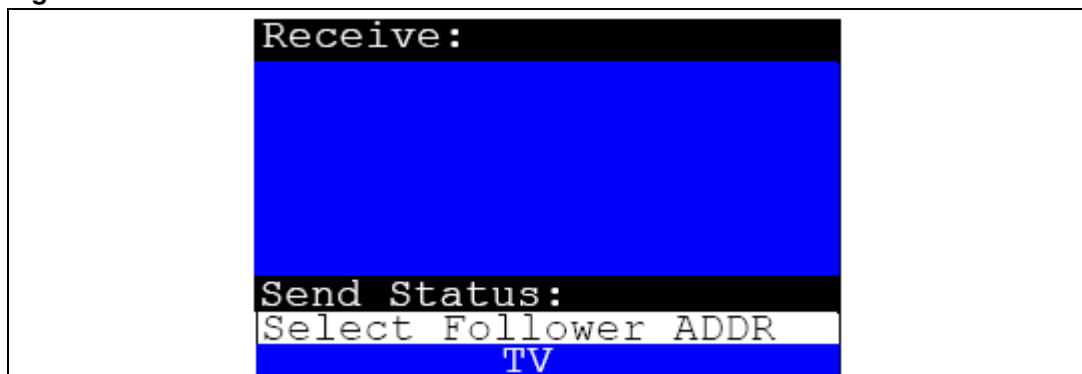
Figure 41. HDMI CEC configuration submenu



The LCD screen is divided into two parts as shown in *Figure 42*:

- A subscreen that shows the CEC receive information: receive status, sender address
- A subscreen that allows to select the follower address and the command to send

Figure 42. CEC menu

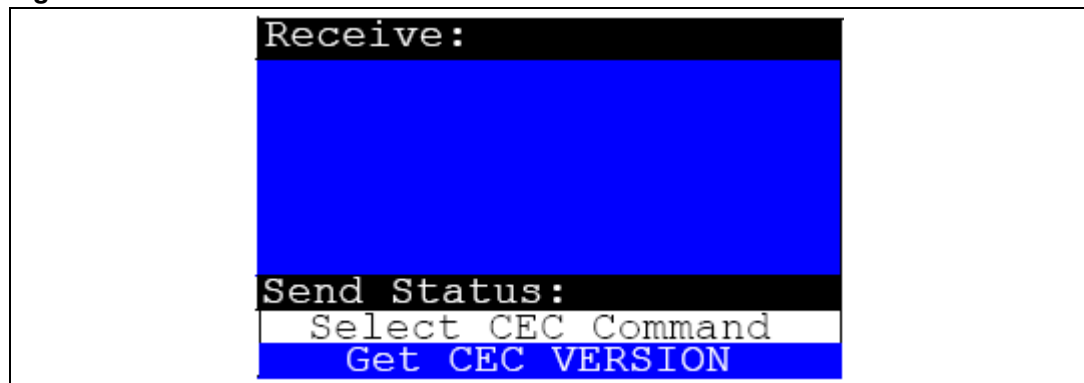


After selecting the follower address, select the command to be sent to the selected follower address using the LEFT, RIGHT and SEL buttons. After selecting the command, the CEC device sends this command to the selected follower address and displays the status of transmission as shown in [Figure 43](#).

You can also send the command from the remote control of the CEC device. After selecting the address, the user presses on the remote control and the message is displayed in the receiver field.

- Note:
- 1 Only, the Phillips RC5 protocol is integrated in the HDMI-CEC application. Each RC5 command has a corresponding HDMI-CEC User Control Code.
 - 2 For more details, refer to [Section 2.4.4: Connectivity submenu](#)

Figure 43. Select CEC command

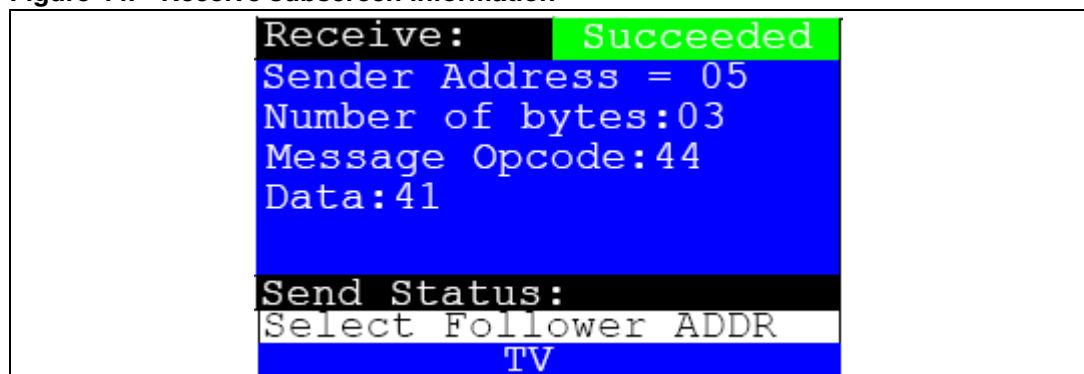


When receiving a new message, the following information can be displayed on the LCD:

- Receive status
- Sender address
- Number of bytes (including the sender address)
- Opcode message
- Data (operands)

[Figure 44](#) shows that the device has correctly received the frame from the sender with address: 0x5, number of bytes received: 0x3 (header + opcode + data), message opcode: 0x44 and data: 0x41

Figure 44. Receive subscreen information



For the command Standby, normally, the device is in stop mode and can wake up only when it receives a new command. But, we keep the Tamper button to get out the HDMI-CEC submenu to avoid the demonstration block.

Any time in the CEC application, If you press the Tamper button, the HDMI CEC stops and you return to the HDMI CEC submenu shown in [Figure 40](#).

Note: The STM32F373VC(T6) CEC device responds only to the following commands. For other commands it sends feature abort:

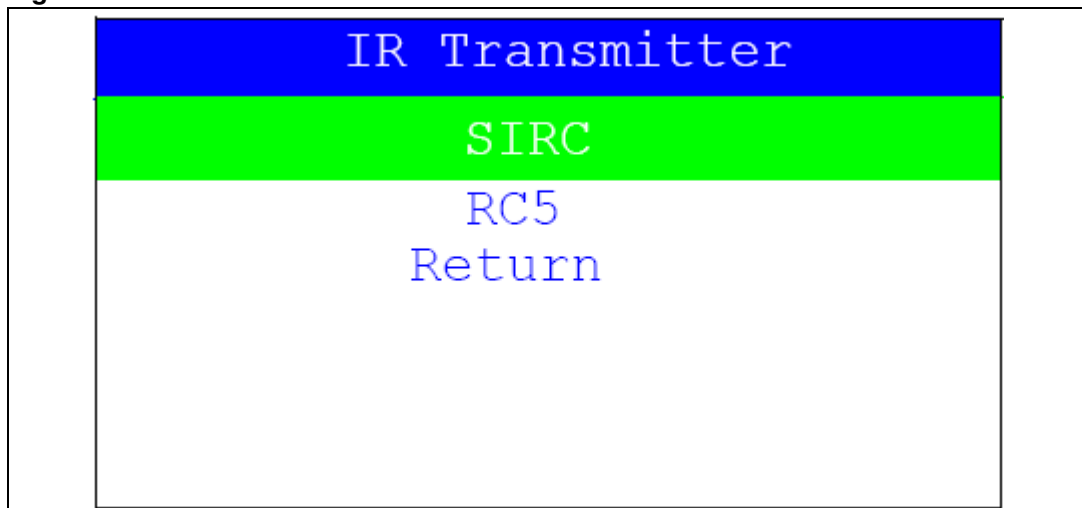
- Standby
- Get CEC version
- Give physical address
- Give OSD name

IR Transmitter

On the STM32373C-EVAL an infra-red LED is driven by PB9 through transistors T6 and T7.

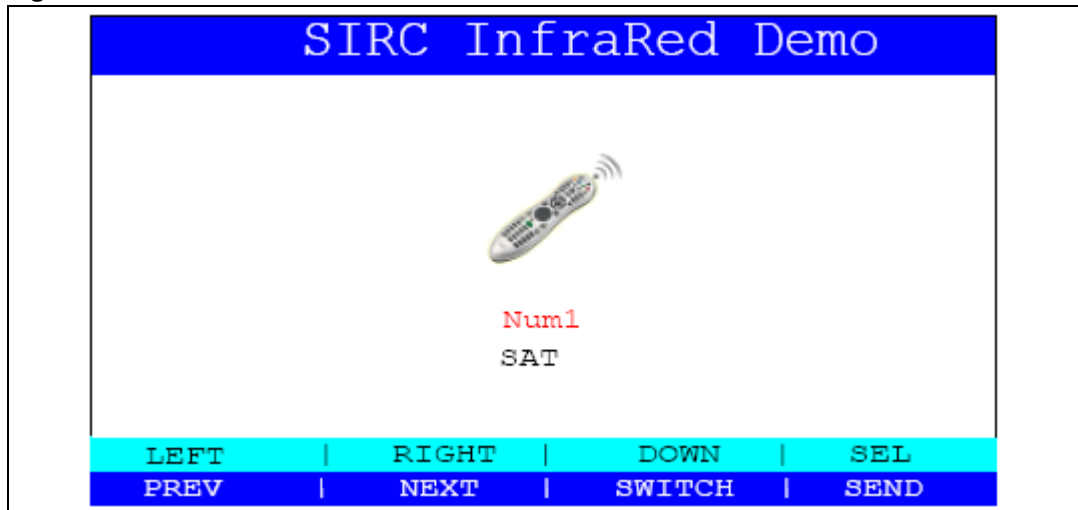
When the IR Transmitter submenu is selected, the message shown in [Figure 45](#) is displayed.

Figure 45. IR transmitter menu



When you select a protocol from the list, the corresponding submenu such as the one shown in [Figure 46](#) is displayed.

Figure 46. IR transmitter command menu



You can control the infra-red transmitter features using the Joystick LEFT, RIGHT, DOWN, UP and SEL buttons:

- Press the joystick DOWN button to switch between device type and command.
- Press the joystick LEFT button to go to the previously defined device/command.
- Press the joystick RIGHT button to go to the next defined device/command.
- Press the joystick SEL button to send the selected device command code.

When the Key is pressed, the infra-red transmitter demo is exited and the menu shown in [Figure 45](#) is displayed.

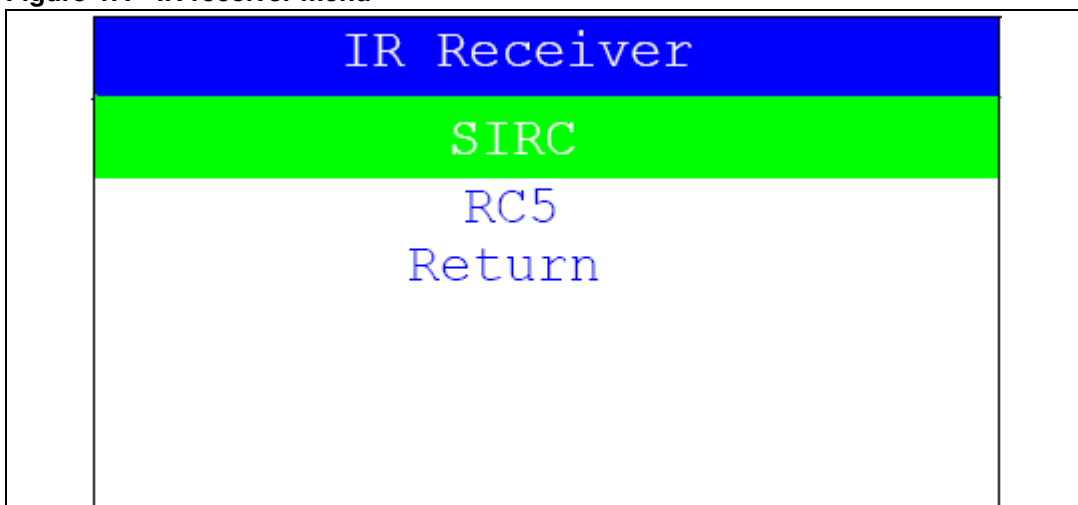
Infra-red receiver

The IR receiver TSOP34836 is connected to PB5 of STM32F373VC(T6) on the STM32373C-EVAL board.

At the receiving end, a receiver detects the light pulses, which are processed to retrieve/decode the information they contain.

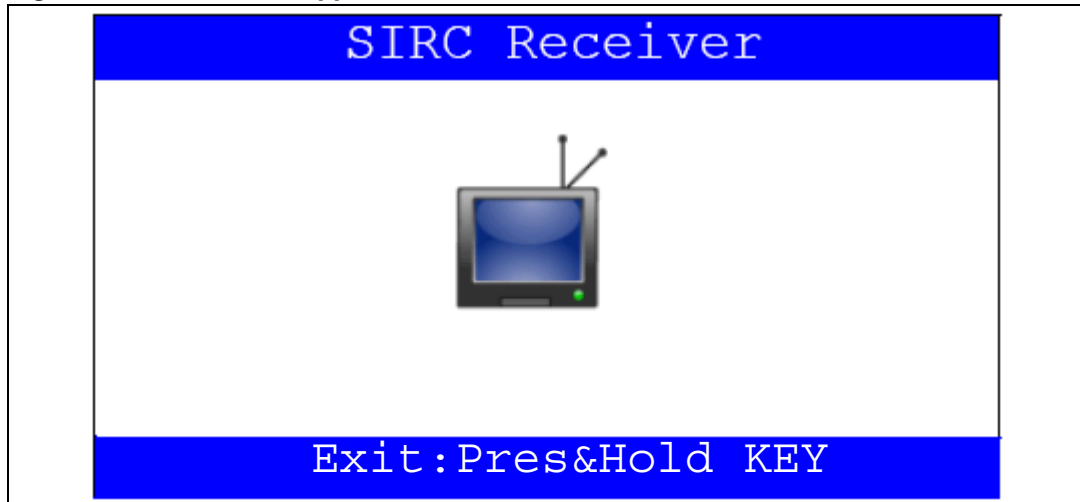
When the IR Receiver submenu is selected, the message shown in [Figure 47](#) is displayed.

Figure 47. IR receiver menu



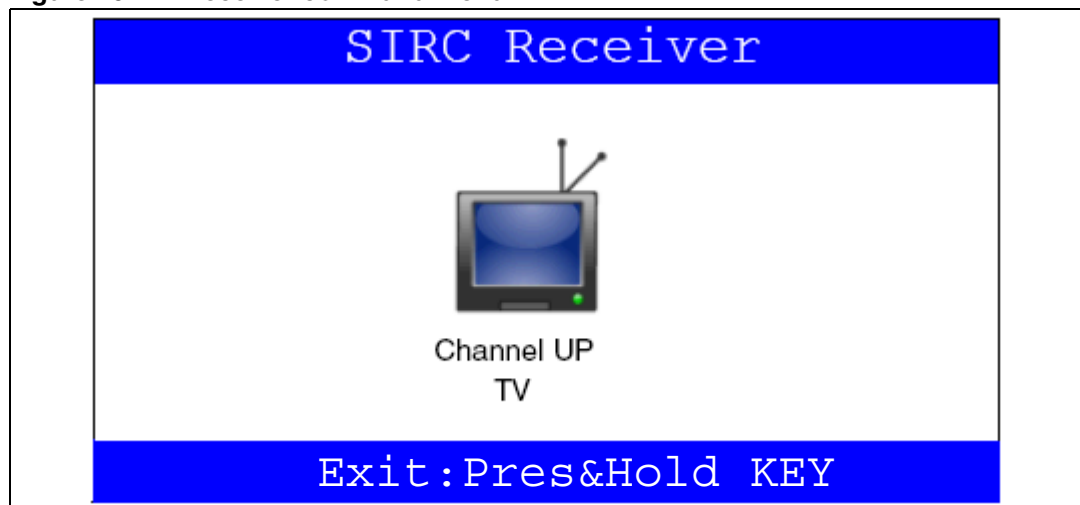
When you select a protocol from the list, the corresponding submenu such as the one shown in [Figure 48](#) is displayed.

Figure 48. IR receiver application menu



When an IR frame is sent using a remote control or using the IR transmitter application running on another STM32373C-EVAL board, this IR frame is decoded using the IR receiver and if the protocol is compatible to the selected IR protocol the device and the command is displayed on the LCD screen [Figure 49](#)

Figure 49. IR receiver command menu



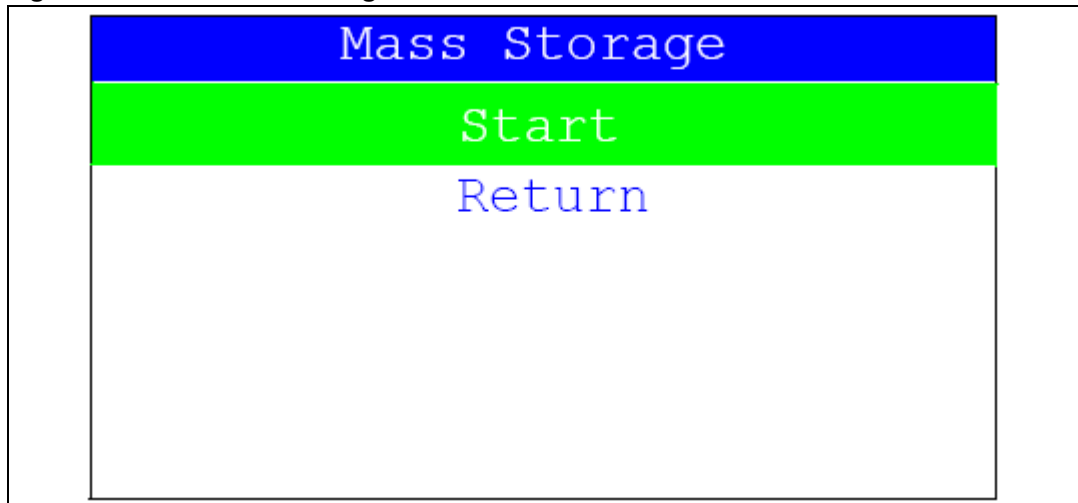
- For more details on the infra-red receiver implementation, refer to application note AN3174: “implementing infra-red remote protocols receiver with the STM32F10xx microcontrollers”. This application note provides a full description of the IR decoding.

USB mass storage

The STM32F373VC(T6) microcontroller features a USB (Universal Serial Bus) that provides a full-speed interface to a USB host PC.

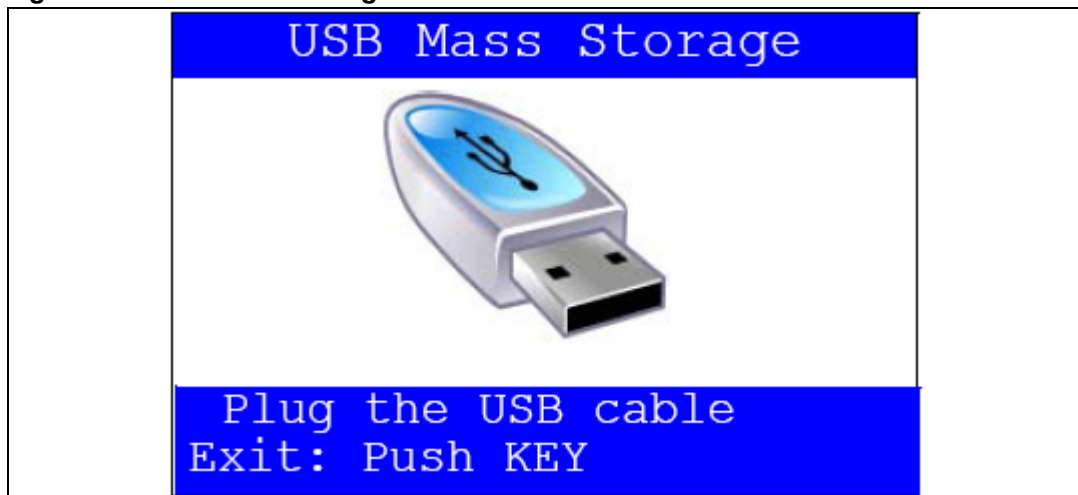
The USB Mass Storage demo is used to configure the USB interface for communication with the PC and to run the mass storage demo using an MSD card.

Figure 50. USB mass storage menu 1



If the SEL push-button is pressed when Start is selected, the message shown in [Figure 51](#) appears on the LCD screen until the cable is plugged in.

Figure 51. USB mass storage menu 2

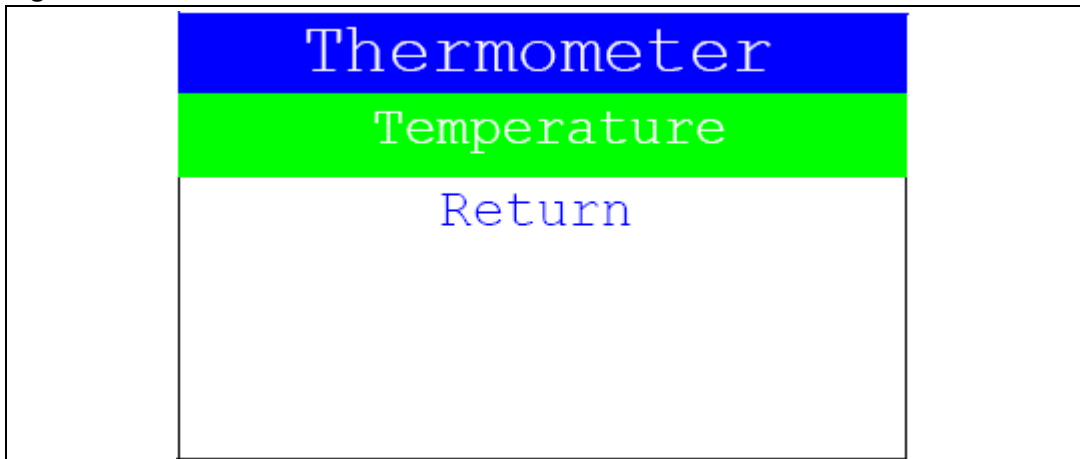


2.4.5 Thermometer

The STM32F373VC(T6) microcontroller has two embedded I²C peripherals that can be connected to any device supporting the I²C protocol including system management bus (SMBus) mode. An STLM75 (or a compatible device) I²C temperature sensor is mounted on the STM32373C-EVAL board and used to capture the external temperature (-55°C to +125°C).

When the Thermometer submenu is selected, the message shown in [Figure 52](#) is displayed on the LCD.

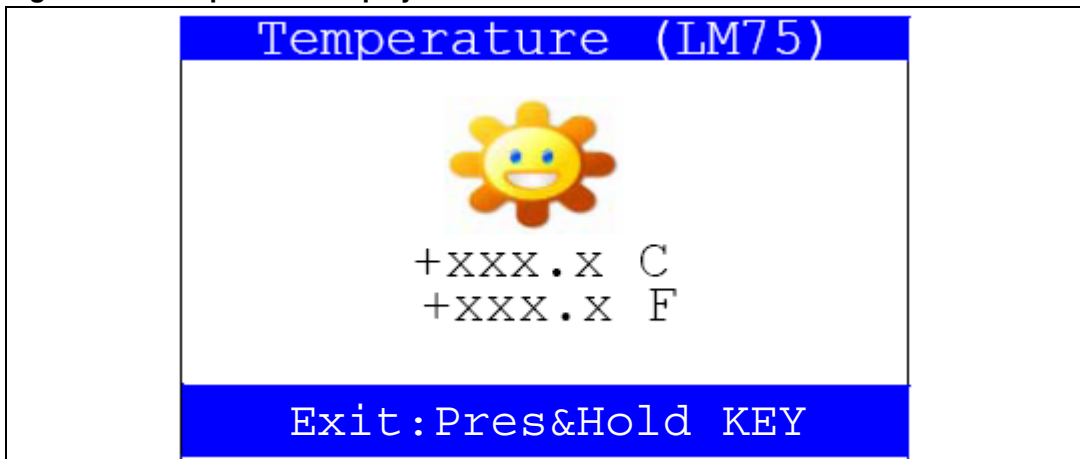
Figure 52. Thermometer submenu selected



Once the Temperature submenu has been selected by pressing the SEL push-button, the temperature value is displayed in Celsius and Fahrenheit as shown in [Figure 53](#).

Press key to return to the Thermometer submenu.

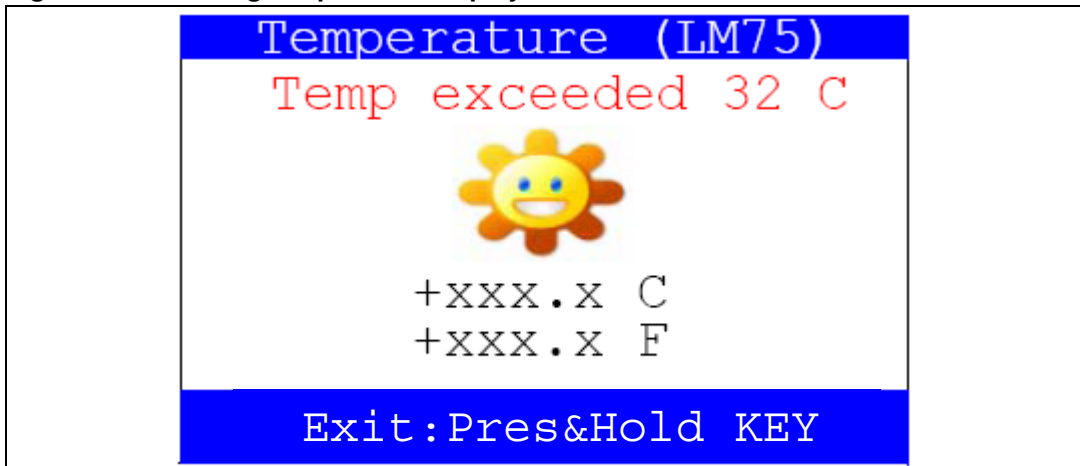
Figure 53. Temperature display



The temperature variations can be monitored easily using the STM32 I2C SMBus feature. This is managed by the SMBus Alert, which generates a dedicated interrupt informing the system that the temperature is out of the selected range. This can be very useful when a higher temperature needs an emergency action, as is the case in critical systems (motor control, medical...).

If the temperature exceeds the high limit (TEMPERATURE_TOS: Over Limit Temperature) the SMBus alert interrupt is generated and the warning message shown in [Figure 54](#) is displayed on the LCD screen.

Figure 54. Warning temperature display



The messages shown in *Figure 53* are displayed on the LCD screen when the temperature goes under the low limit (TEMPERATURE_THYS: Hysteresis Temperature).

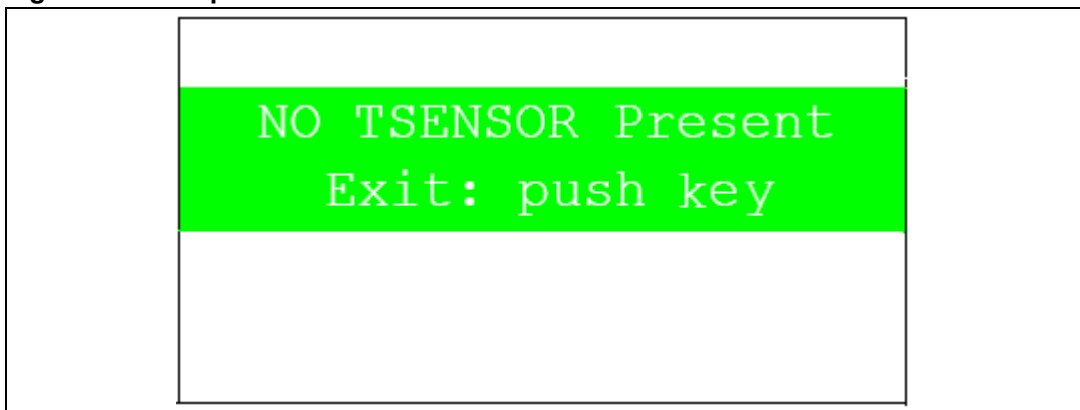
The user can configure the TOS and THYS using dedicated define values in the code. By default, the STM32373C-EVAL demo sets them to the value defined in the thermometer.c file:

```
#define TEMPERATURE_THYS 31
#define TEMPERATURE_TOS 32
```

Press KEY to return to the Thermometer submenu.

Note: Any hardware problem with the temperature sensor is detected by a test. In such case, the message shown in *Figure 55* is displayed.

Figure 55. Temperature sensor error



2.4.6 Low-power modes

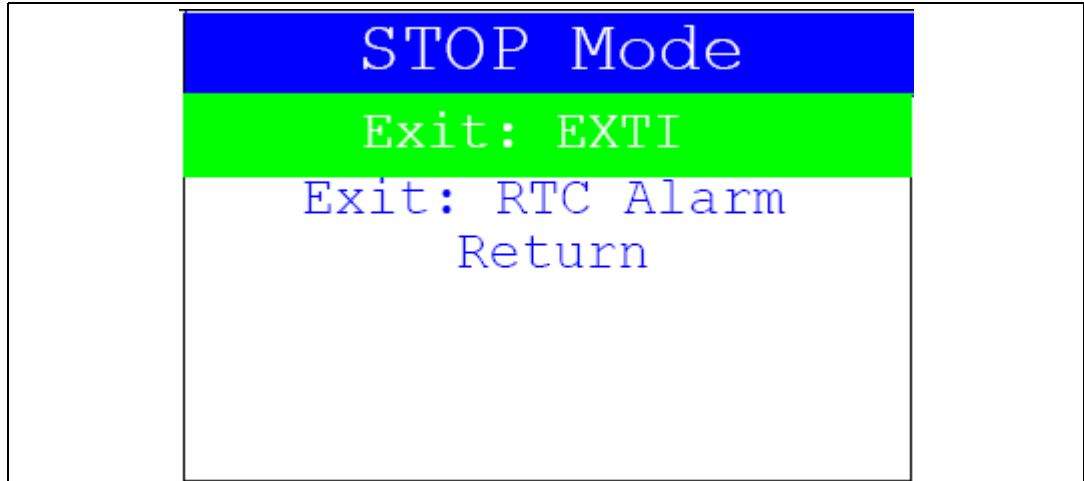
The STM32F373VC(T6) microcontroller provides different operating modes in which the power consumption is reduced. The purpose of this menu is to show the behavior of the microcontroller in different low-power modes. Stop and Standby modes are taken as examples.

Stop mode menu

This menu allows the user to put the STM32F373VC(T6) in Stop mode. The firmware performs the specific instruction sequence needed to enter Stop mode.

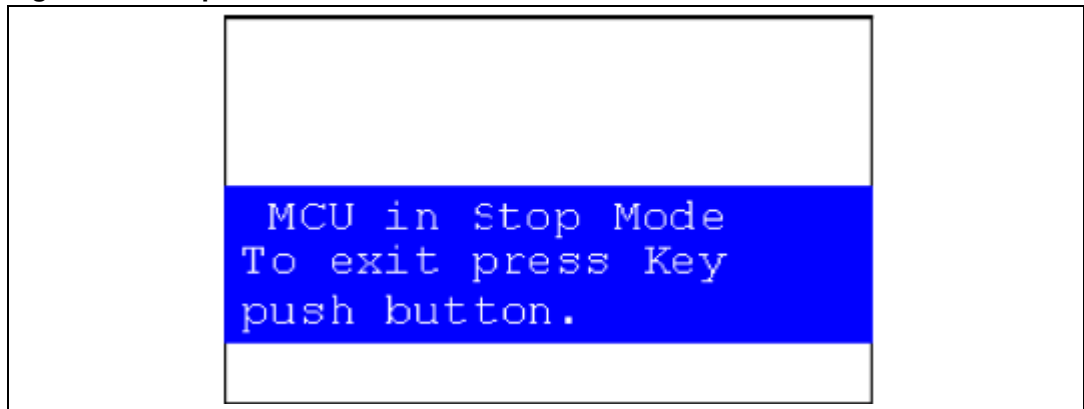
In this application, the STM32F373VC(T6) can exit Stop mode in two ways as shown in [Figure 56](#).

Figure 56. Exiting Stop mode



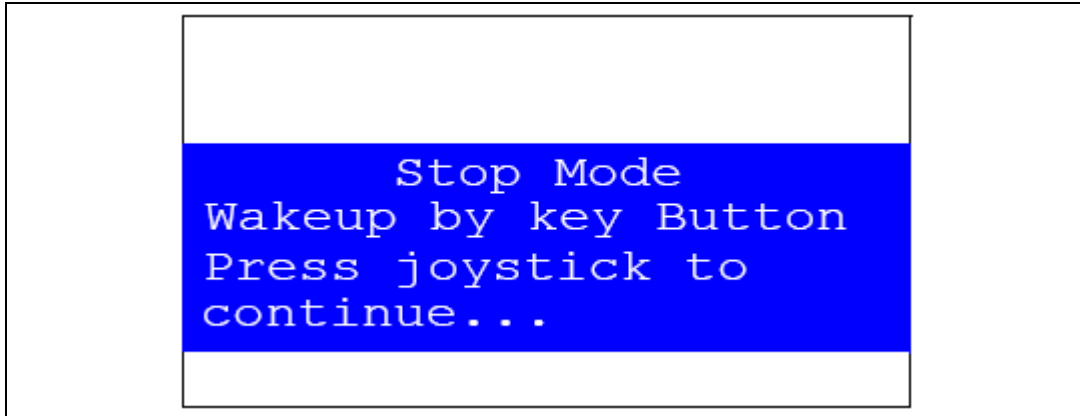
- In the first case, The EXTI Key button is used to make the MCU exit Stop mode. Once the **Stop mode** submenu has been selected, the four LEDs continue blinking until the “SEL” push-button is pressed, and the system enters Stop mode. When the MCU is in Stop mode, the message shown in [Figure 57](#) is displayed on the LCD.

Figure 57. Stop mode entered exit EXTI

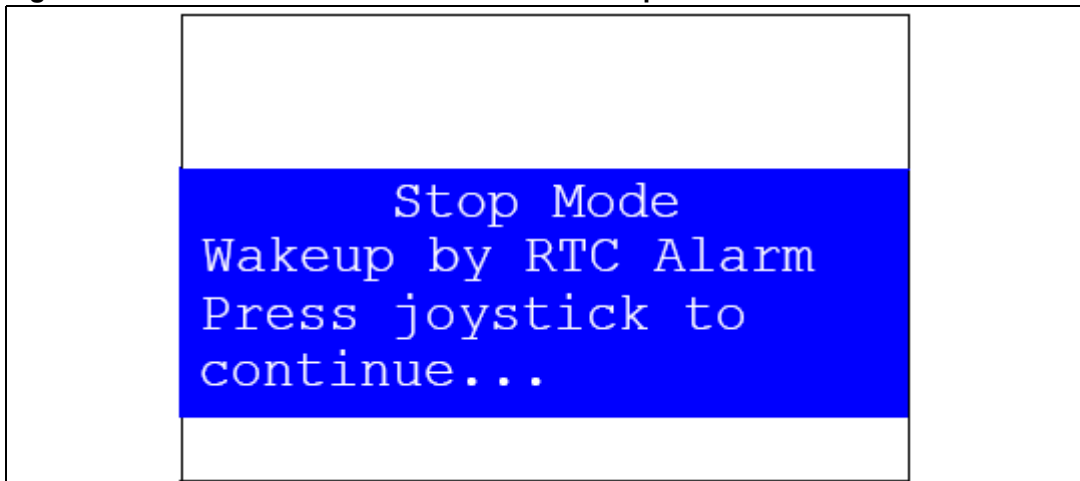


The MCU remains in the Stop mode until the Key push-button is pressed and the message shown in [Figure 58](#) is displayed on the LCD screen.

Once the Key push-button has been pressed, the MCU exits the Stop mode. The system clock is then set to 72 MHz and the application resumes execution.

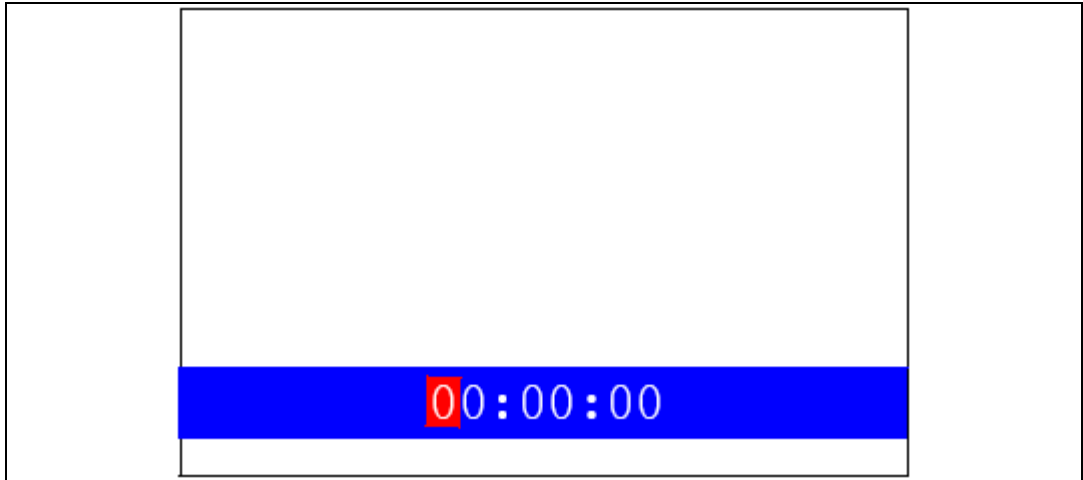
Figure 58. MCU in the Stop mode Exit EXT1

Note: If an RTC Alarm is generated while the MCU is in Stop mode and the message shown in [Figure 58](#) is displayed (which means that the Key push-button needs to be pressed to exit Stop mode), the RTC Alarm causes the MCU to exit Stop mode. The message shown in [Figure 59](#) is then displayed.

Figure 59. RTC Alarm causes the MCU to exit Stop mode

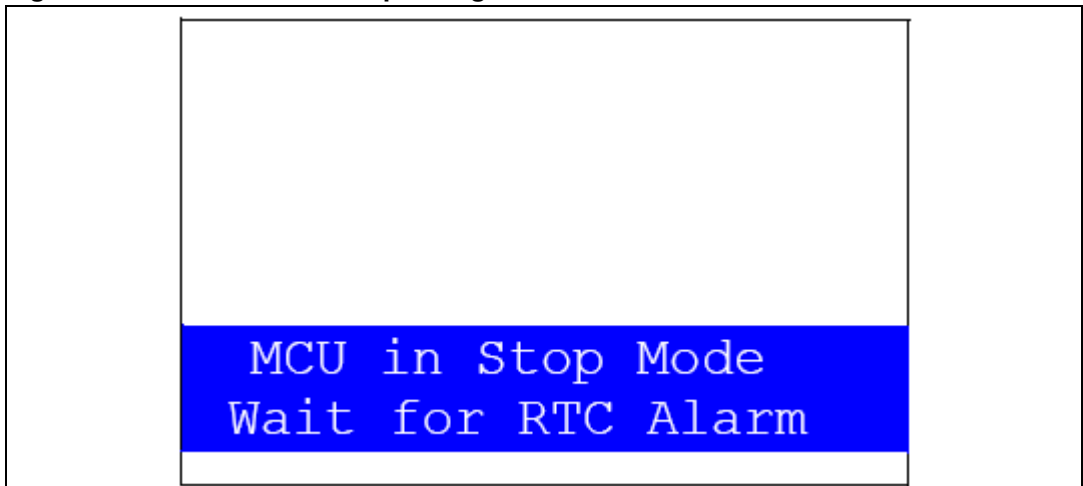
- In the second case, the RTC Alarm wakes up the MCU from Stop mode after the programmed time has elapsed. When selecting this submenu, the user has to set the alarm to the time when the MCU is to exit Stop mode. [Figure 60](#) shows how to set the wakeup time.

Figure 60. Setting the wakeup time



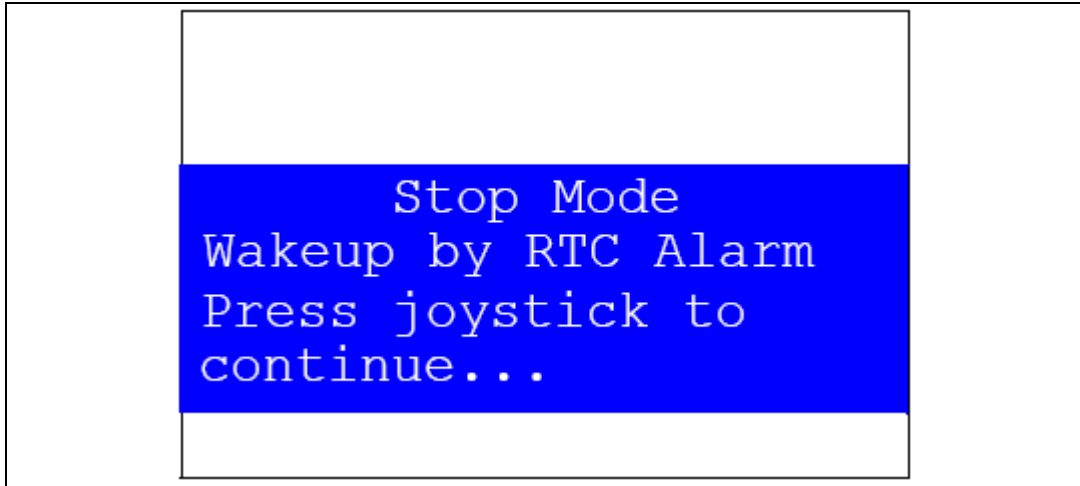
Once the alarm has been configured, the four LEDs stop blinking and the system enters Stop mode. The message shown in [Figure 61](#) is displayed on the LCD.

Figure 61. RTC Alarm wakeup configured



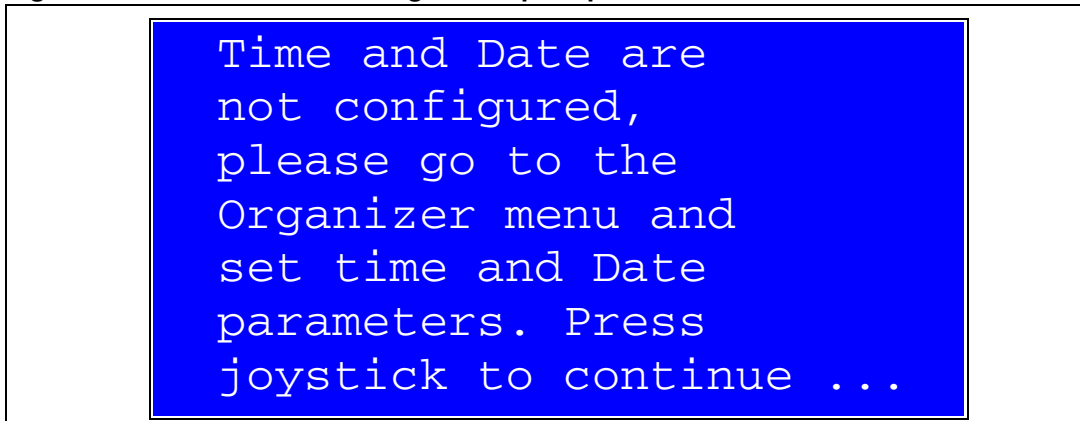
After the programmed time has elapsed, the system exits Stop mode. The system clock is then set to 72 MHz and the application resumes execution. The message shown in [Figure 62](#) is displayed on the LCD screen.

Figure 62. RTC Alarm wakeup



Note: If the Time and Date have not been set, the message shown in [Figure 63](#) is displayed on the LCD screen.

Figure 63. Time and date configuration prompt

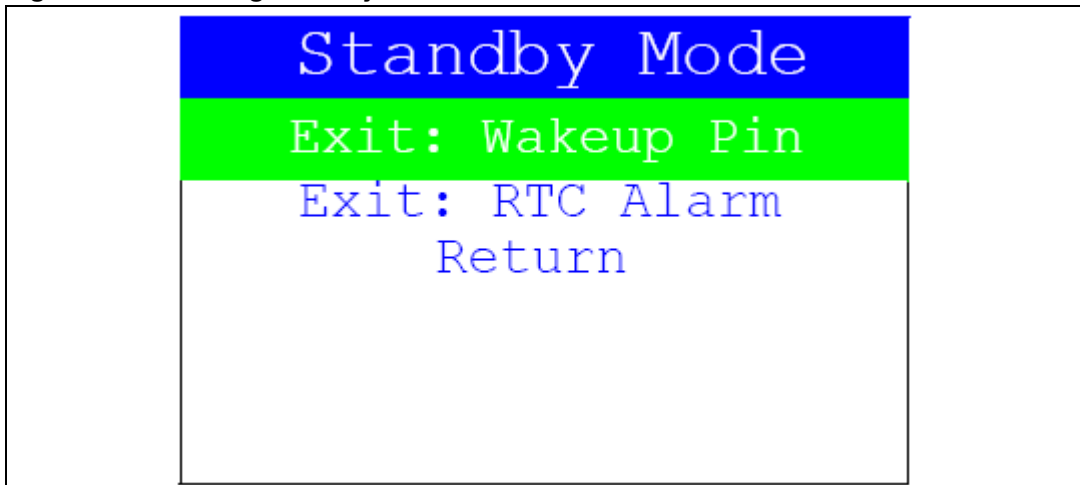


Standby mode menu

This menu allows the user to put the STM32F373VC(T6) in Standby mode. The software runs the specific instruction sequence needed by the STM32F373VC(T6) to enter Standby mode.

In this application, the STM32F373VC(T6) can be made to exit Standby mode in two ways as shown in [Figure 64](#).

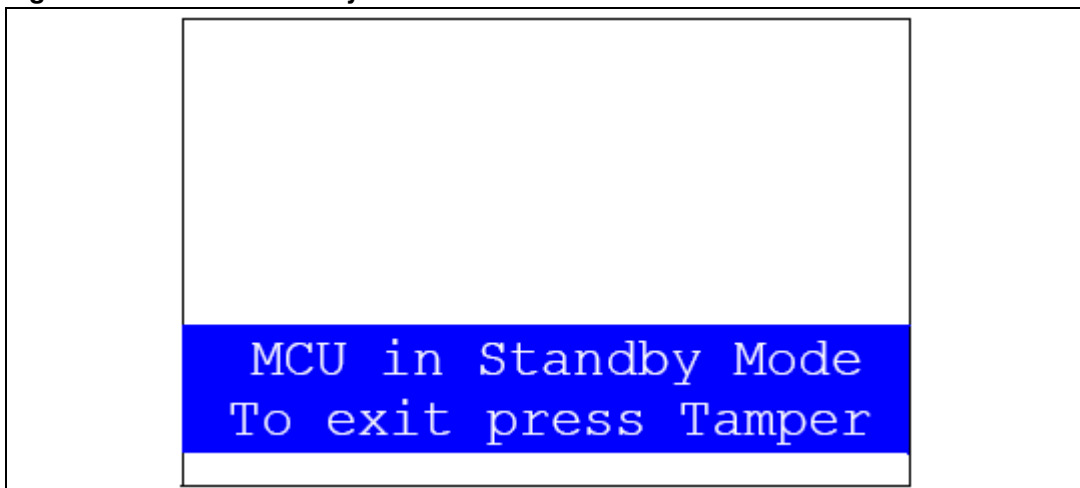
Figure 64. Entering Standby mode



- In the first case, the Wakeup push-button is used to wake up the MCU from Standby mode.

Once the **Standby mode** submenu has been selected, the four LEDs continue blinking until the "SEL" push-button is pressed, and the system enters Standby mode. When the MCU is in Standby mode, the message shown in [Figure 65](#) is displayed on the LCD

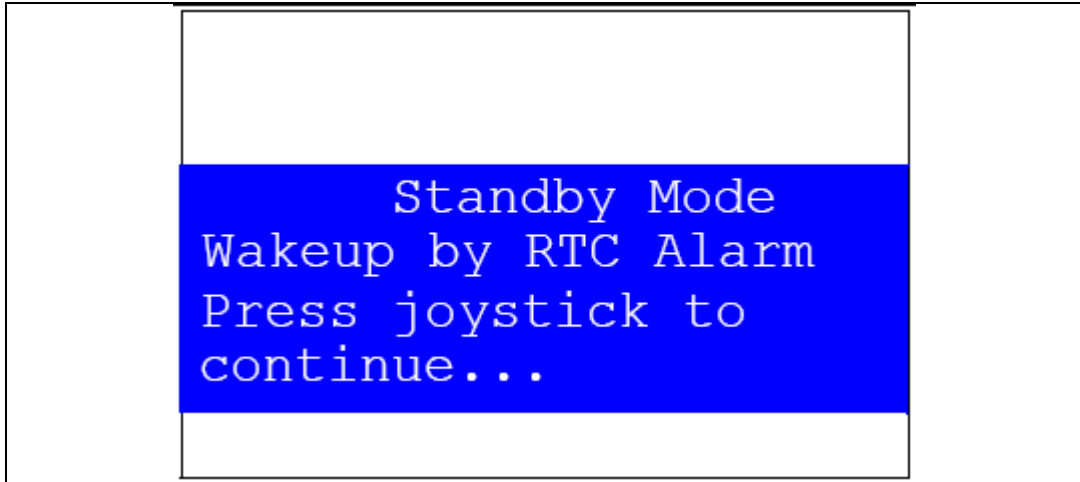
Figure 65. MCU in Standby mode



The MCU remains in Standby mode until the tamper push-button is pressed. Once the Wakeup push-button has been pressed, the MCU exits Standby mode and the system reset signal is generated.

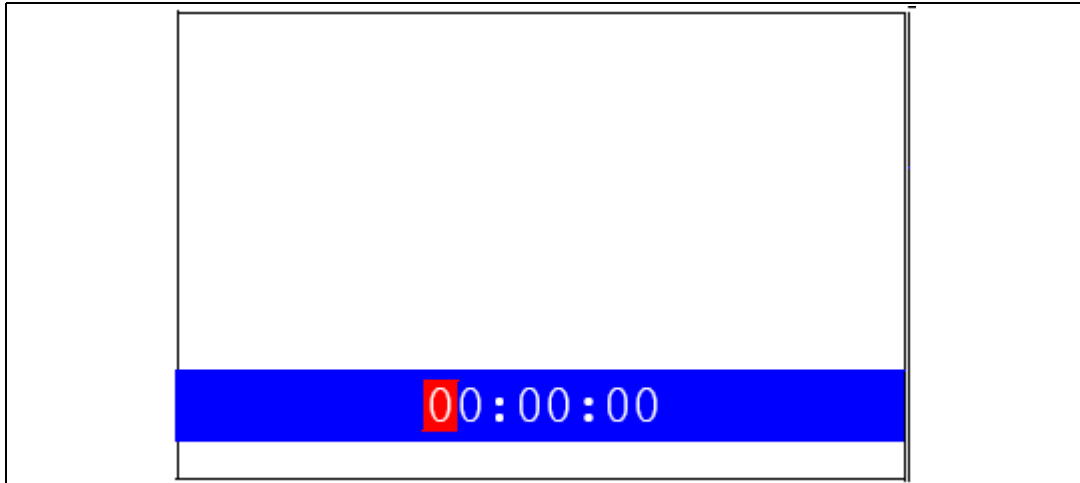
Note: If an RTC Alarm is generated while the MCU is in Standby mode and the message shown in [Figure 66](#) is displayed (which means that the tamper push-button needs to be pressed to exit Standby mode), the RTC Alarm causes the MCU to exit Standby mode and a system reset signal is generated.

Figure 66. RTC Alarm causes the MCU to exit the Standby mode

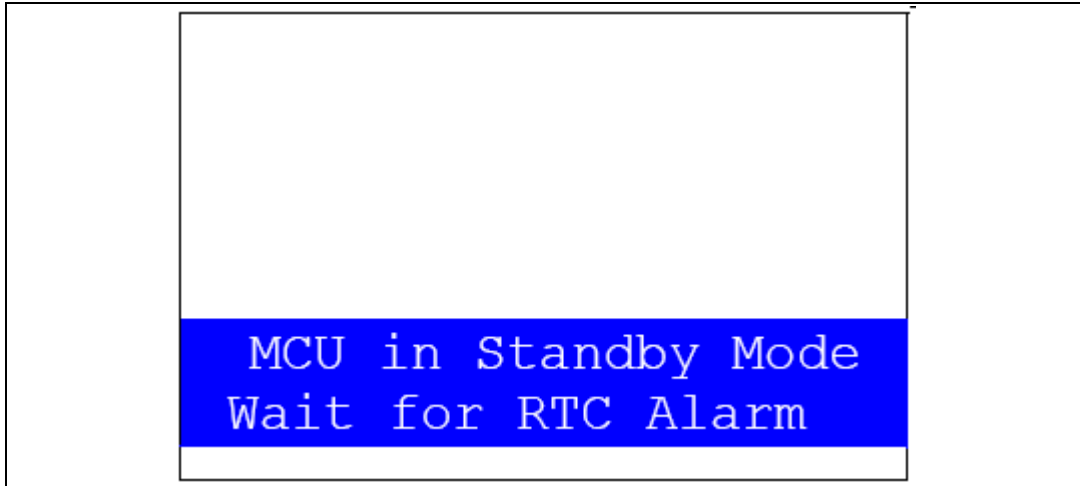


- In the second case, the RTC Alarm wakes up the MCU from the Standby mode after the programmed time has elapsed. When selecting this submenu, the user has to set the alarm to the time when the MCU is to exit the Standby mode. [Figure 67](#) shows how to set the wakeup time.

Figure 67. Setting the wakeup time

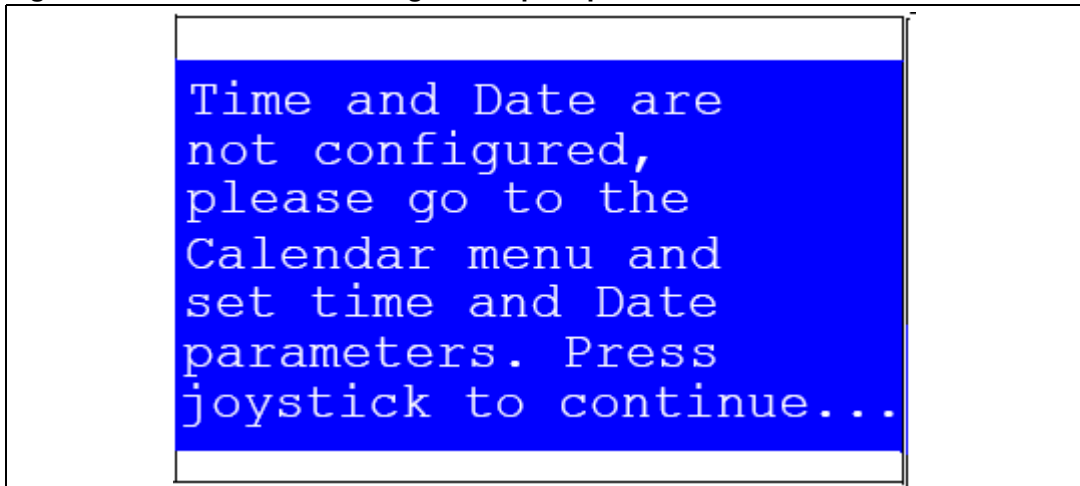


Once the alarm has been configured, The LEDs stop blinking and the system enters Standby mode. The message shown in [Figure 68](#) is then displayed on the LCD.

Figure 68. RTC Alarm wakeup configured

After the programmed timing has elapsed, the system exits Standby mode and a system reset signal is generated.

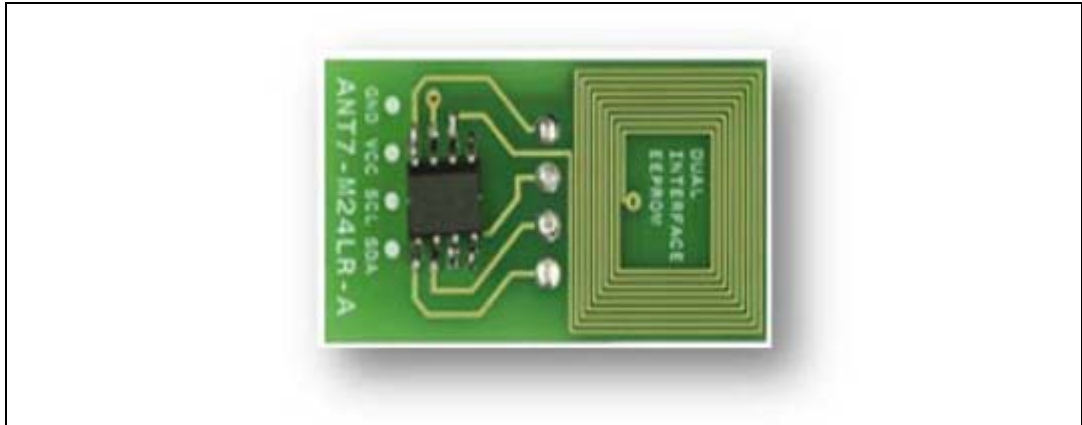
Note: if the Time and Date have not been set, the message shown in [Figure 69](#) is displayed on the LCD screen.

Figure 69. Time and date configuration prompt

2.4.7 RF EEPROM

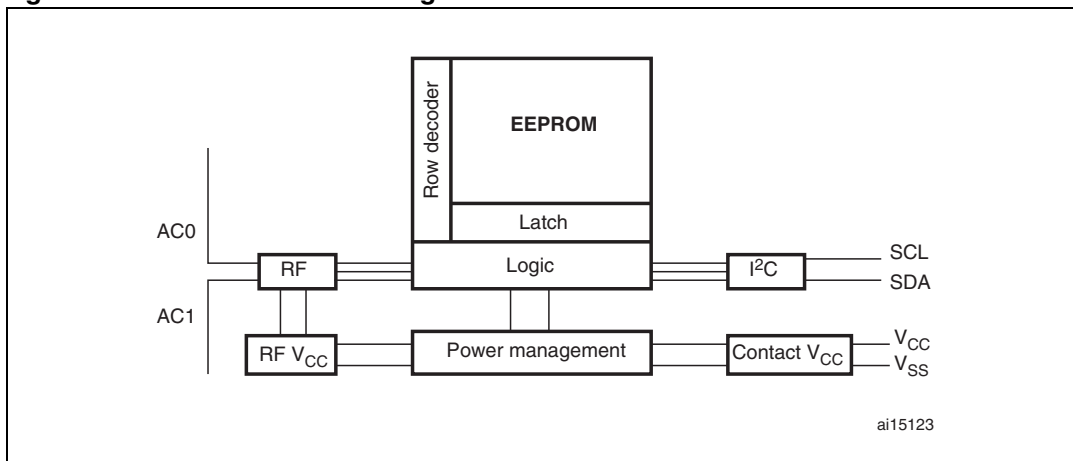
The STM32F373VC(T6) microcontroller has two embedded I²C peripherals that can be connected to any device supporting the I²C protocol. An RF EEPROM daughter board can be connected to STM32373C-EVAL via the I²C interface.

Figure 70. RF EEPROM daughter board



The M24LR64-R device is a dual-interface, electrically erasable programmable memory (EEPROM). It features an I²C interface and can be operated from a V_{CC} power supply. It is also a contactless memory powered by the received carrier electromagnetic wave. The M24LR64-R is organized as 8192 × 8 bits in I²C mode and as 2048 × 32 bits in ISO 15693 and ISO 18000-3 mode 1 RF modes.

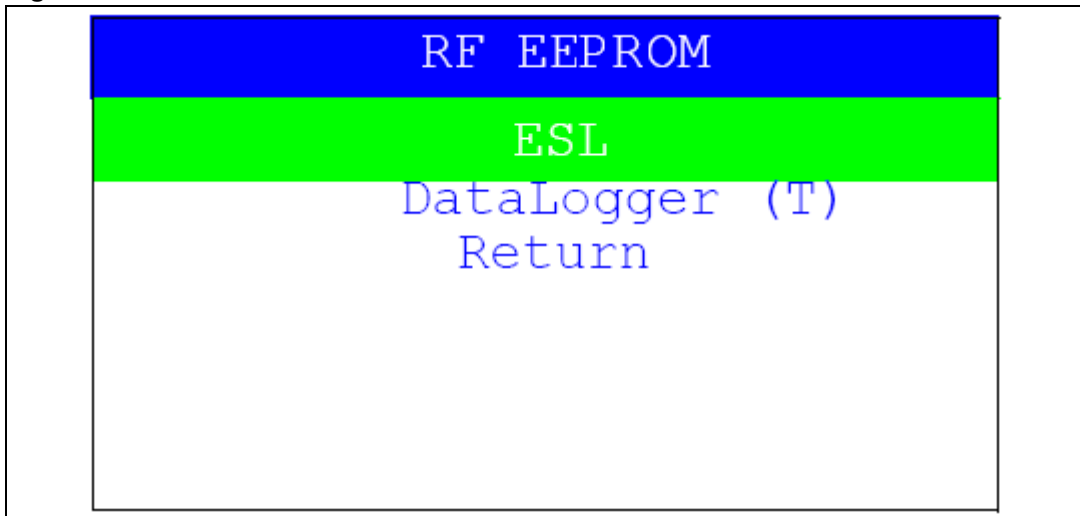
Figure 71. M24LR64-R block diagram



Note: For more details on the M24LR64 EEPROM, please refer to the datasheet.

Select the RF EEPROM menu by pressing SEL from the main menu, two RF EEPROM applications can be selected as shown in [Figure 72](#).

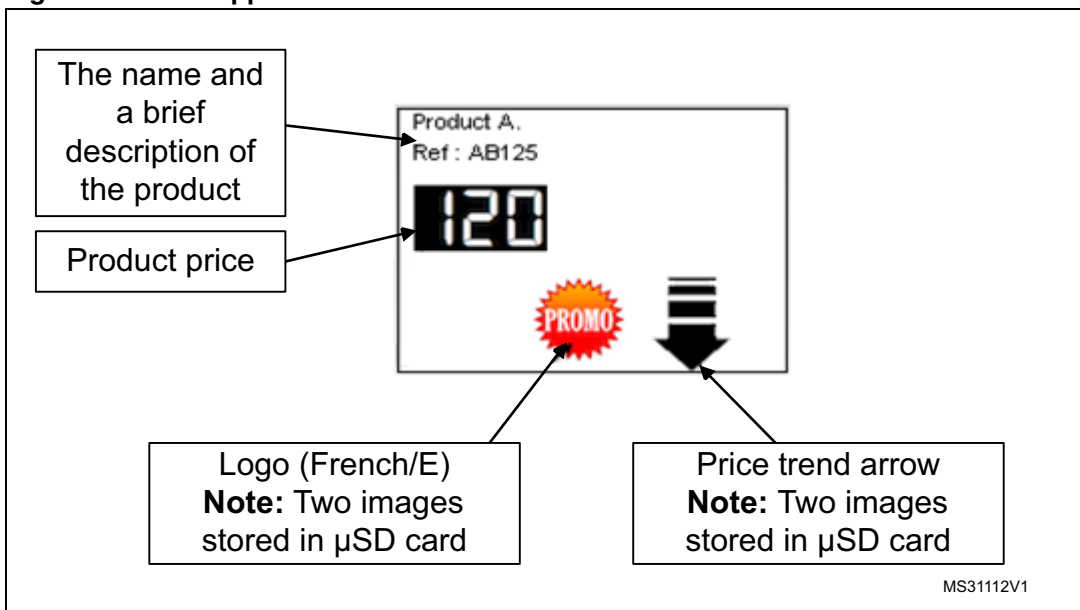
Figure 72. RF EEPROM menu



ESL application

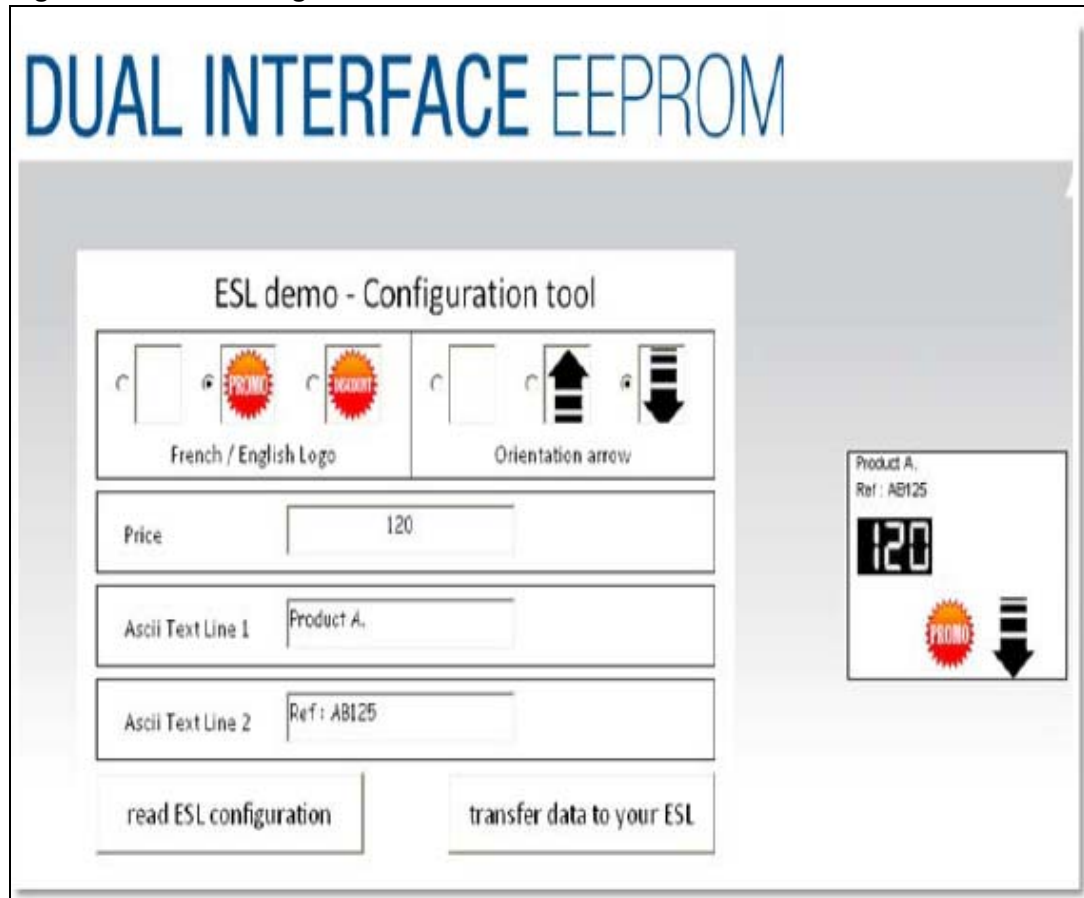
The Electronic Shelf Labeling (ESL) application consists of reading the content of the RF EEPROM (using I2C interface of the M24LR64 dual interface EEPROM) and displaying the information on the LCD screen as shown in *Figure 73*.

Figure 73. ESL application



All this information (product name, price, logo, price trend...) are programmed in the M24LR64 dual interface EEPROM via the RF interface and can be updated any time using the CR95HF demo board with the associated PC software (**M24LRxx_Application_Software**).

Figure 74. ESL setting menu



Using this PC software application, several parameters can be set to configure the ESL device and update the LCD display:

Logo: Check the French (PROMO) or English (DISCOUNT) to indicate that a special price is proposed. Check the blank logo if no special price is applied.

Price trend arrow: Check the up or down arrow to indicate if the price has been increased or decreased, or the blank arrow if no indication is required.

Product price

ASCII Text line 1 and Line2: Display the name and a brief description of the product.

Note: For more details about this PC Software please refer to user manual **UM0853**.

Using this PC software, the M24LR64 dual interface EEPROM is organized as shown in [Table 5](#).

Table 5. M24LR64 memory organization (ESL application)

RF block address	bits [31:24]	bits [23:16]	bits [15:8]	bits [7:0]
0x0000		Arrow	Icon	
0x0001	Contains 0xXXXXXXXX X = number of digits in price (max 7) YYYYYYYY = price value including decimal separator (.) coded in hexadecimal. For instance 0x512A99FF means 5 digits and price is 12.99			
0x0002	Contains the ASCII codes of the string chain called ASCII TEXT LINE1 in M24LR64 ESL demo software. The maximum size of this text line is 20 characters.			
0x0003				
0x0004				
0x0005				
0x0006				
0x0007	Contains the ASCII codes of the string chain called ASCII TEXT LINE2 in M24LR64 ESL demo software. The maximum size of this text line is 20 characters.			
0x0008				
0x0009				
0x000A				
0x000B				
0x000C	XXXX XX is the checksum of bytes from address 0x0000 to 0x000B. YY = Reserved ZZ = Reserved			

DataLogger application

This RF EEPROM application demonstrates a practical application of the M24LR64. It is able to record and store 64 Kbits of temperature data using the M24LR64-R Dual interface EEPROM (I2C and RF). The DataLogger microcontroller is an STM32F373VC(T6). It communicates with the M24LR64-R using its serial interface and controls an STLM75 digital temperature sensor.

Figure 75. Communication block diagram

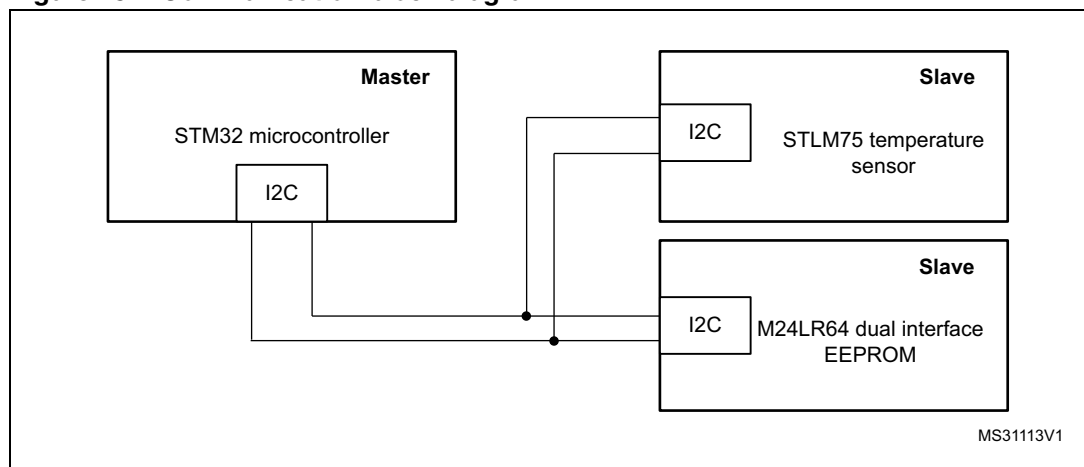
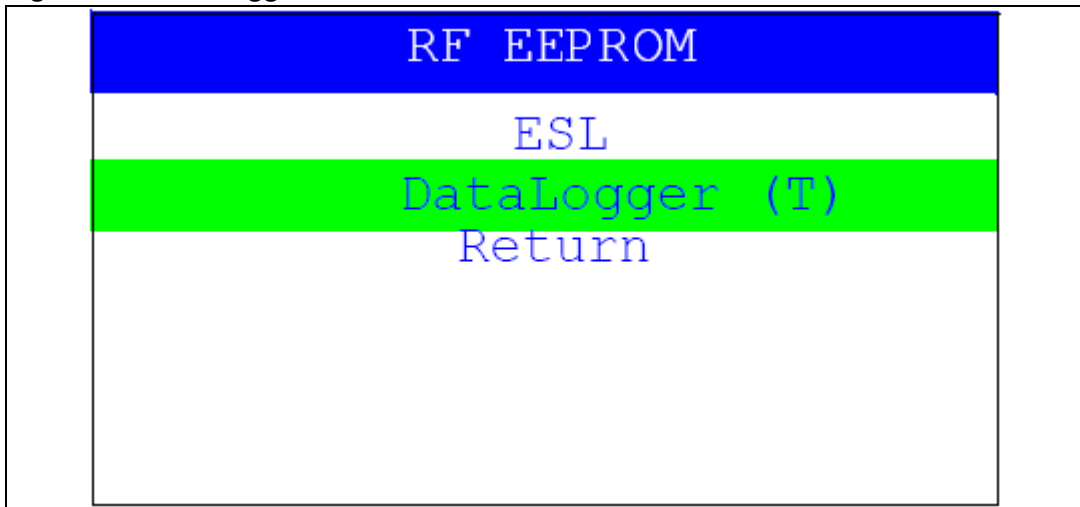


Figure 76. DataLogger sub-menu



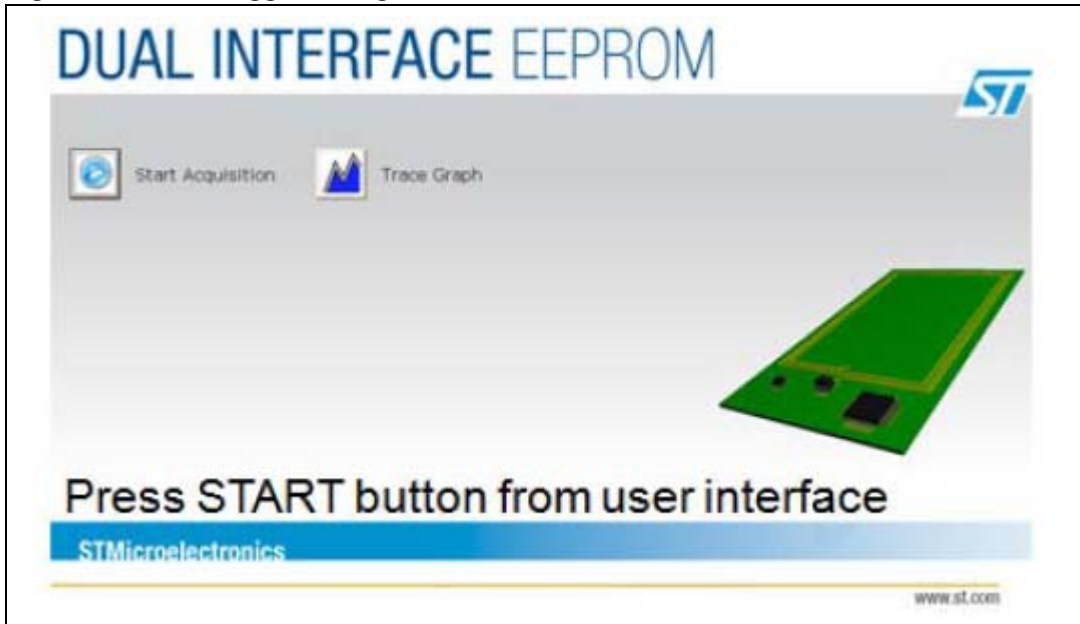
An on-board demonstration firmware stored in the STM32F373VC(T6) memory consist to selects, controls the temperature acquisition through a RFID reader connected by a USB cable to a PC.

For more details about this application please refer to the application note AN3109 “Developing your M24LR64-R DataLogger application for temperature acquisition”.

How to run the application:

1. From the STM32F373C evaluation board side, select the DataLogger submenu and then press SEL push button.

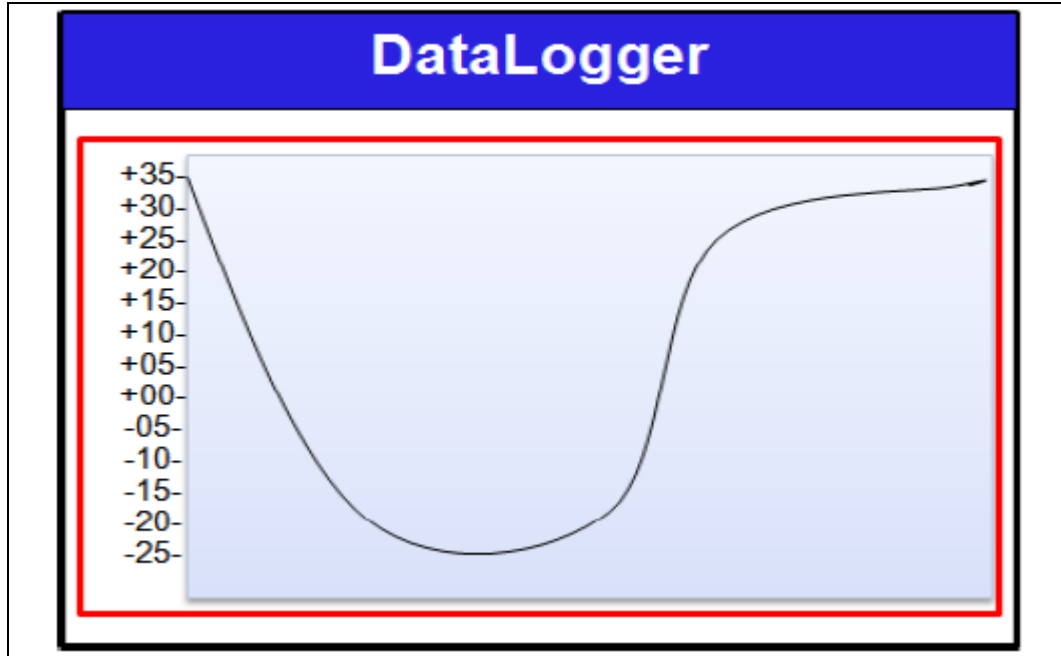
Figure 77. DataLogger dialog box



2. From the PC software side, choose “DataLogger” application and then follow the steps described on the user manual **UM0925**.

When the application starts, the data is stored in the dual interface EEPROM and the temperature data is displayed in a real time as a graph on the TFT-LCD display on the evaluation board and also on the PC software interface.

Figure 78. DataLogger curve



Reference documents:

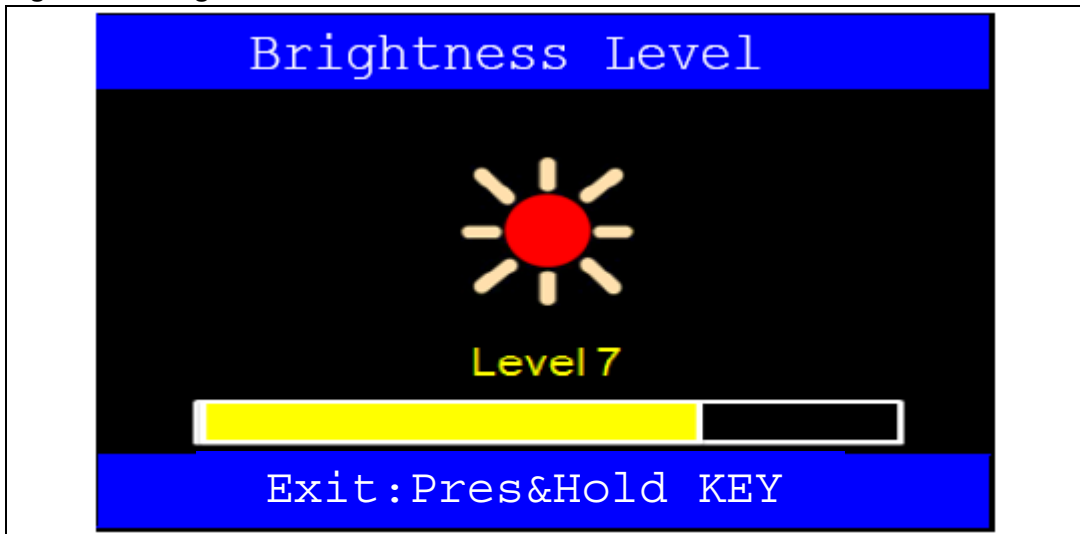
- a) M24LR64-R datasheet
- b) M24LR64-R tool driver install guide user manual (UM0863)
- c) User manual UM0925: *Using the M24LR64-R DataLogger reference design*
- d) Application note AN3057: *How to manage M24LR64-R data transfers from the I²C bus or an RF channel*
- e) Application note AN3109: *Communication peripheral FIFO emulation with DMA and DMA timeout in STM32F10x microcontrollers*

2.4.8 Brightness application

The STM32F373VC(T6) microcontroller has an embedded comparator which can be used in combination with DAC peripheral for different applications like detecting brightness level (using the LDR sensor).

When the LDR submenu is selected, the message shown in [Figure 79](#) is displayed.

Figure 79. Brightness level

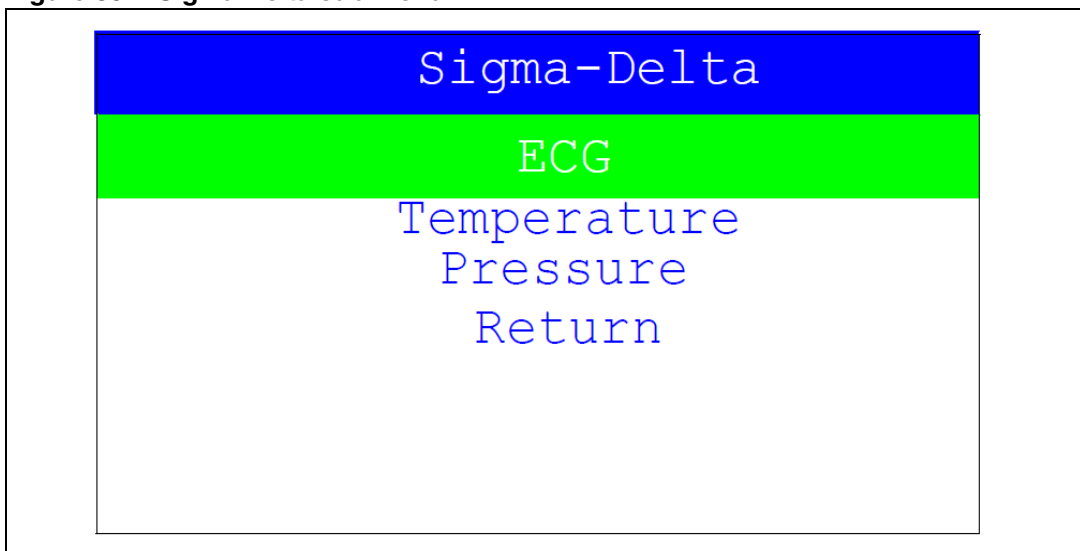


Depending on the brightness level, the icons and the progress bar are changed.

2.4.9 Sigma-Delta applications

A precise 16-bit sigma-delta ADC (SDADC) is embedded in the STM32F373VC(T6) microcontroller for use in various applications. In the STM32373C-EVAL demo firmware package, three sub-demos are available to show practical applications of the SDADC. The figure below shows the sigma delta sub-demo and the available applications available: Electrocardiogram acquisition, temperature measurement using a PT100 sensor and pressure measurement using an MPX2102A differential sensor.

Figure 80. Sigma-Delta sub-menu



Electrocardiogram sub-demo

Two ECG electrodes TS1 and TS2 are available on the STM32373C-EVAL board for connecting to the user's right and left fingers. Two amplification stages are available

between the STM32F373VC(T6) device and the ECG electrodes. The first stage gain is set to 5 and the second one is set to 33.

Note: The second stage gain can be set to 133 when jumper JP1 is installed.

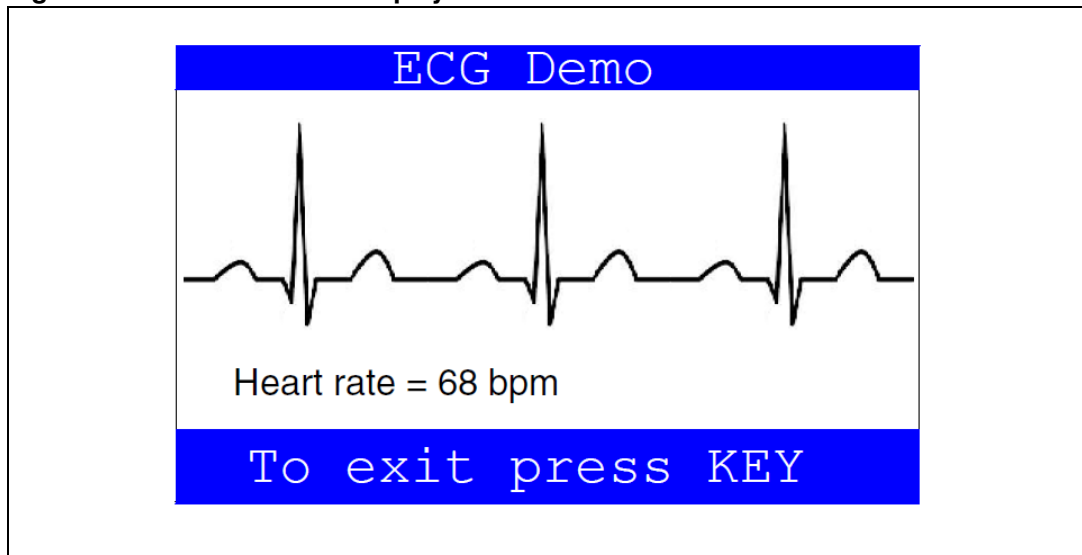
The 16-bit SDADC, triggered by a timer, is used to convert the analog signal and the DMA (double buffering) for data transfer. A bandpass FIR filter is used to filter out noise from useful ECG data. Once filtered the ECG data are displayed on the TFT-LCD display of the ST32373C-EVAL board. The heartbeat rate per minute is displayed on the LCD.

Note: It is recommended to run the demo with JP1 fitted.

Caution: Recommended conditions for ECG detection and heartbeat measurement are:

1. Humid air and fingers
2. Large area in contact with the electrodes
3. Relaxed body with no movement
4. Third electrode usage connected to GND
5. Evaluation board preferably powered by USB
6. Body must be electrically isolated from earth

Figure 81. ECG waveform display



Temperature sub-demo

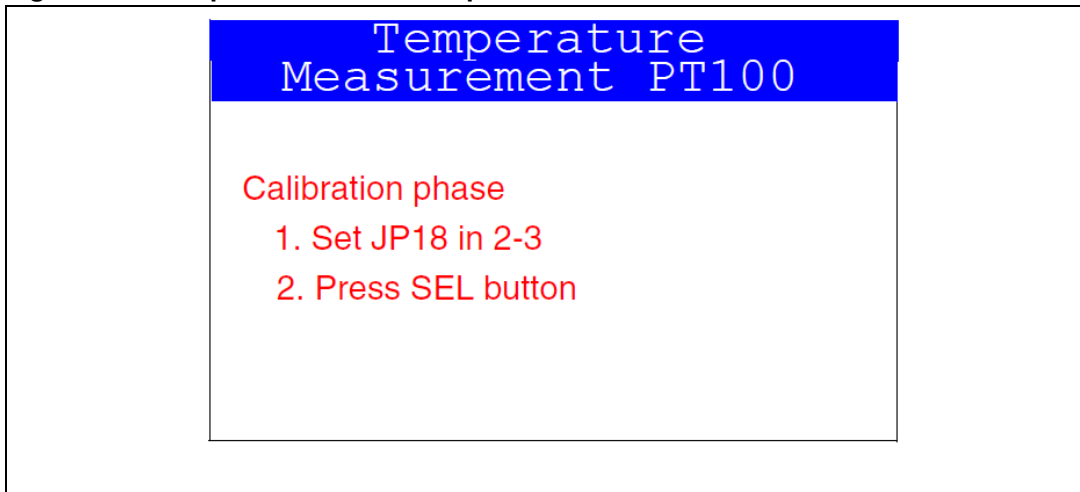
A current source circuit is available on the STM32373C-EVAL board to provide a fixed 1 mA current to a platinum probe (PT100). The output voltage level is connected to the Sigma delta ADC in STM32F373VCT6 through PE7 to measure temperature value on PT100.

The temperature measurement is performed in two steps:

1. Temperature sensor calibration: this phase is performed with JP18 installed in 2-3 (REF) position. A 100 Ohm resistor is used for calibration.

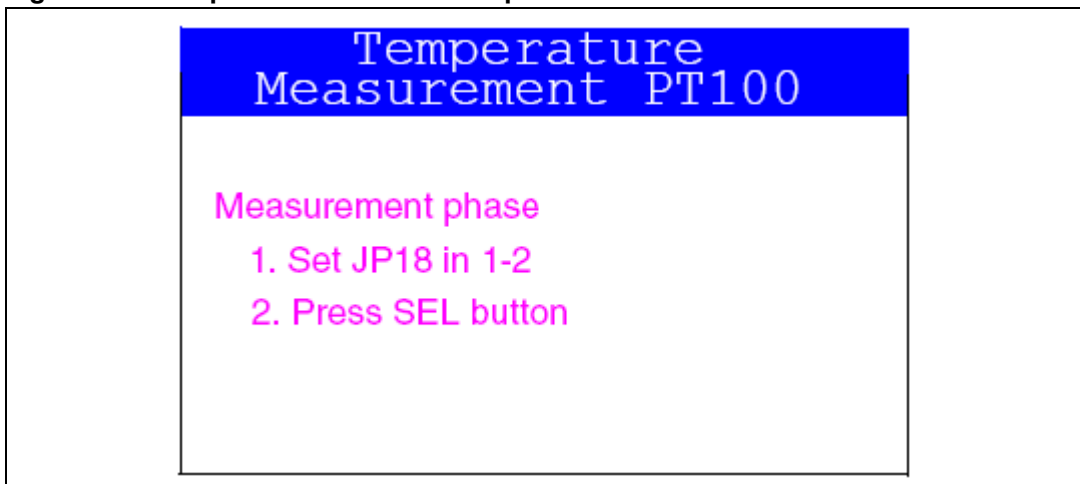
In the calibration phase, a correction factor is computed and saved in backup registers. If you want to run the calibration phase again, power-off the evaluation board and set the jumper JP11 in VDD position. If jumper JP11 is fitted in BAT position, the calibration phase is executed only once.

Figure 82. Temperature calibration phase



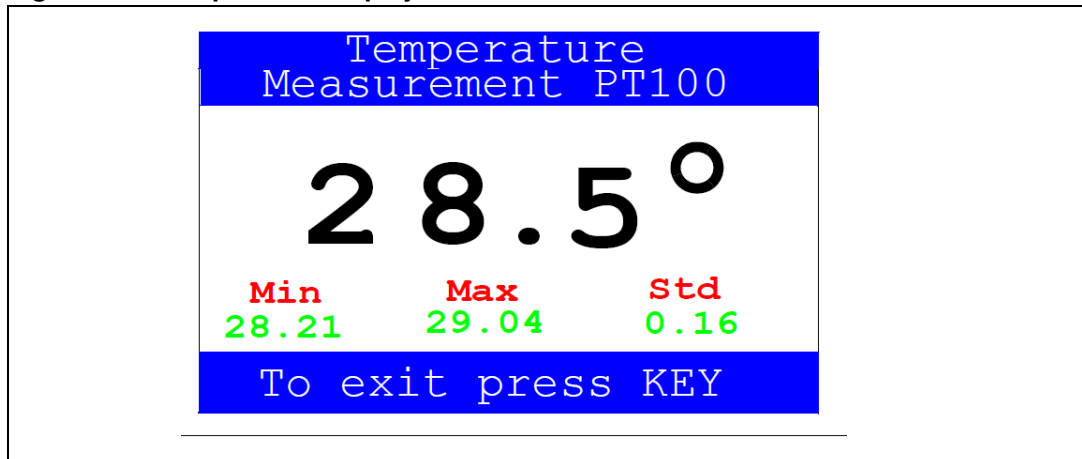
- 2. The temperature measurement: this phase is performed with JP18 installed in position 1-2 (PT100) position.

Figure 83. Temperature measurement phase



The average value of 256 samples, the minimum, the maximum and the standard deviation is displayed on the TFT-LCD. Samples are displayed in Celsius degrees. The figure below shows an example of the messages displayed by the temperature sub-demo on the TFT-LCD.

Figure 84. Temperature display



Pressure sub-demo

An MPX2102A 1000HPa full scale absolute pressure sensor and analog front end are provided on the STM32373C-EVAL board. The output differential pair is connected to the Sigma delta ADC of the STM32F373VCT6 via the PE8 (P) and PE9 (N) pins.

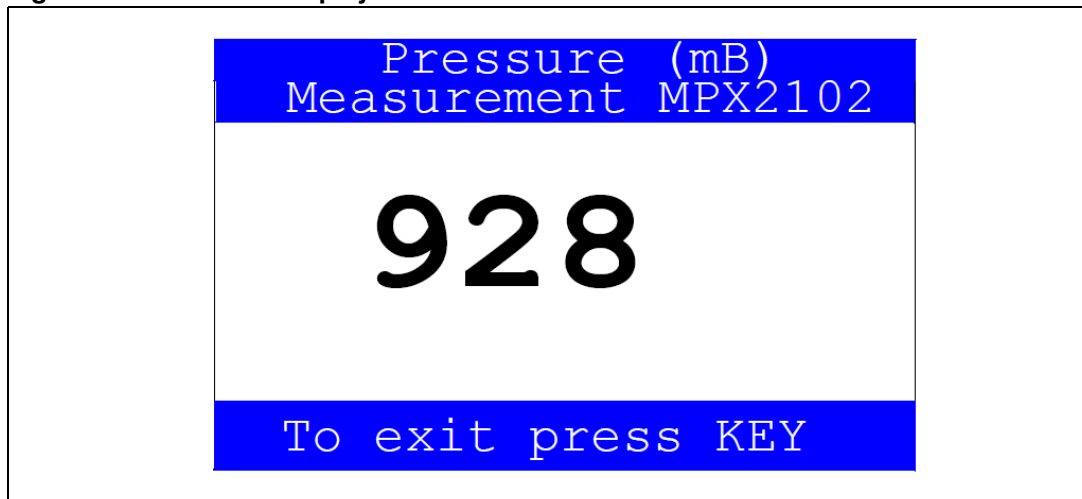
The pressure sub-demo allows you to calibrate the pressure when a known pressure is available. You provide the current pressure using a reference source. Otherwise the pressure sub-demo displays the measured pressure.

The calibration factor is saved in backup registers. If you want to run the calibration phase again, power-off the evaluation board and set the jumper JP11 in VDD position.

If jumper JP11 is fitted in BAT position, the calibration phase is executed only once.

Figure 85 below displays the measured pressure in millibar (mB)

Figure 85. Pressure display



2.4.10 Help

This submenu provide help for configuring the jumpers on the STM32373C-EVAL evaluation board and navigating between the menus and submenus available in the demo firmware.

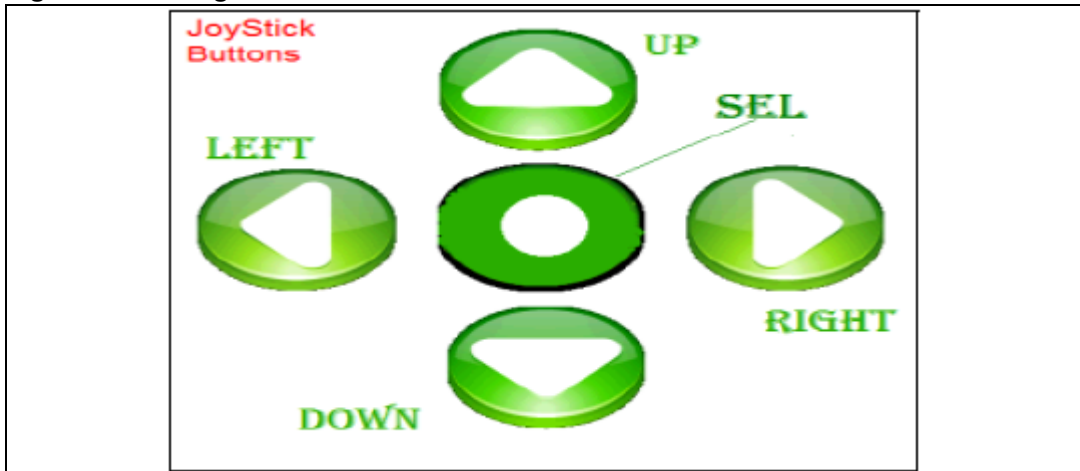
Select the Help menu by pressing SEL from the main menu. The message shown in [Figure 86](#) is then displayed on the LCD screen.

Figure 86. Help menu

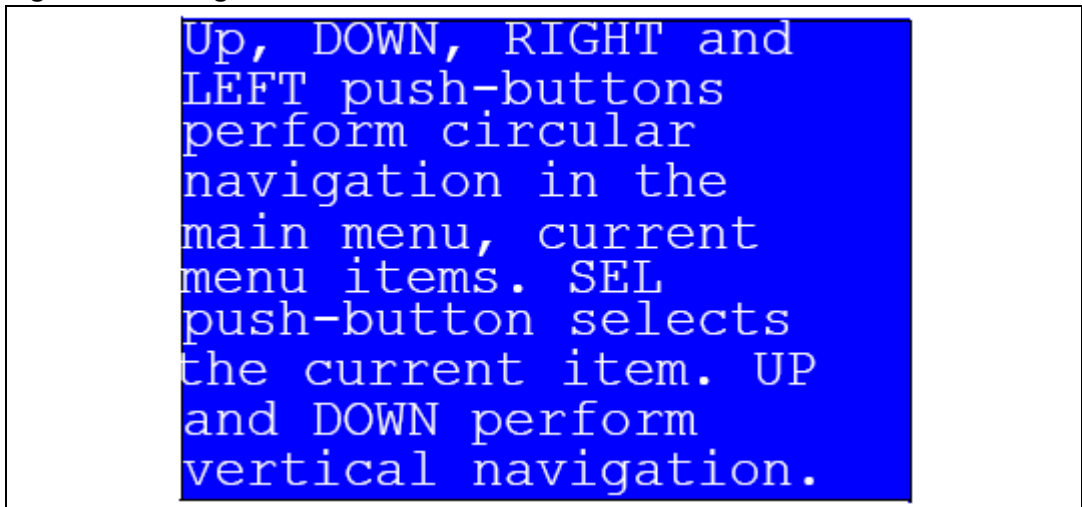


- If the Menu Navigation submenu is selected, the message shown in [Figure 87](#) is displayed.

Figure 87. Navigation menu 1

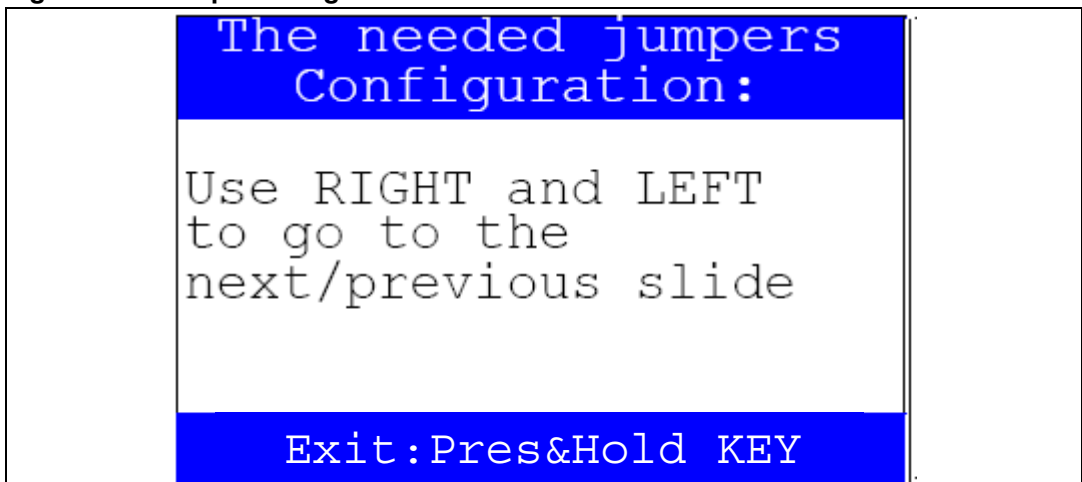


When any joystick button is pressed, the second navigation interface is displayed as shown in [Figure 88](#).

Figure 88. Navigation menu 2

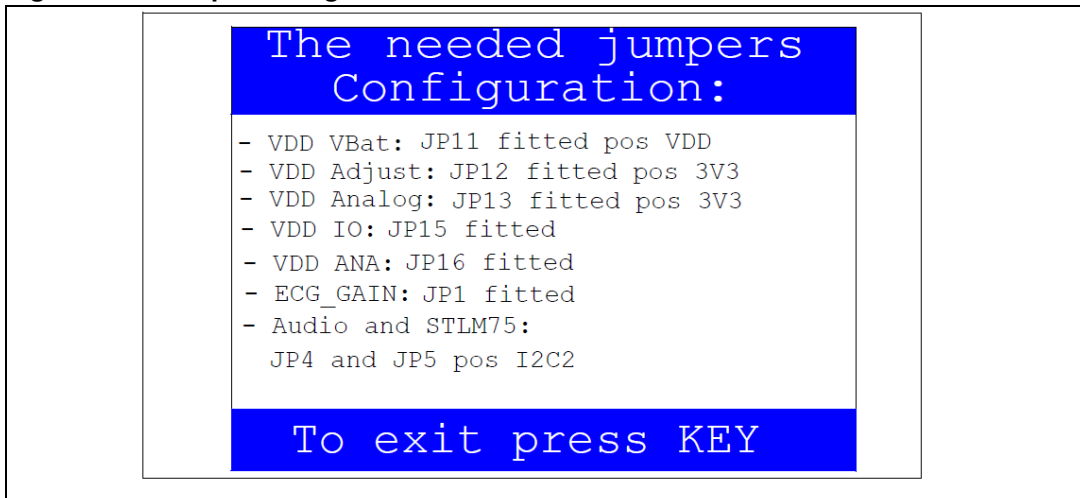
Once the joystick push-button has been pressed again, the MCU exits the navigation submenu and the message shown in [Figure 86](#) is displayed on the LCD screen.

- If the Jumpers config submenu is selected, the message shown in [Figure 89](#) is displayed.

Figure 89. Jumper configuration menu 1

Press RIGHT and LEFT to go to the next/previous slide and UP to exit. The following messages shown in [Figure 89](#), [Figure 90](#) and [Figure 91](#) are displayed sequentially on the LCD screen.

Figure 90. Jumper configuration menu 2



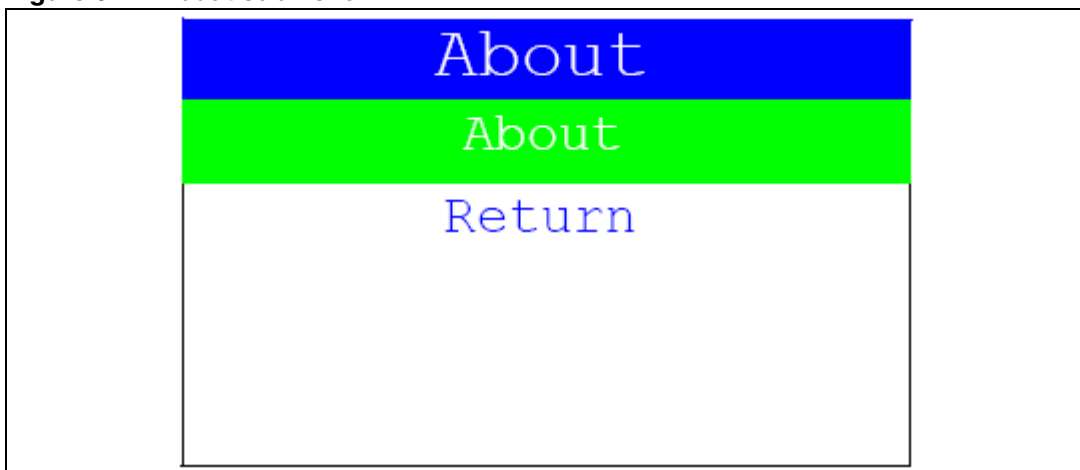
Set JP11 in BAT position to save time even if the evaluation board is powered off. By default, JP11 is set in position VDD.

Once the KEY push-button has been pressed, the MCU exits the Jumper configuration submenu and the message shown in [Figure 86](#) is displayed on the LCD screen.

2.4.11 About submenu

This submenu shows the version of the STM32F373VC(T6) demo firmware. When the About submenu is selected, the message shown in [Figure 91](#) is displayed on the LCD screen.

Figure 91. About submenu

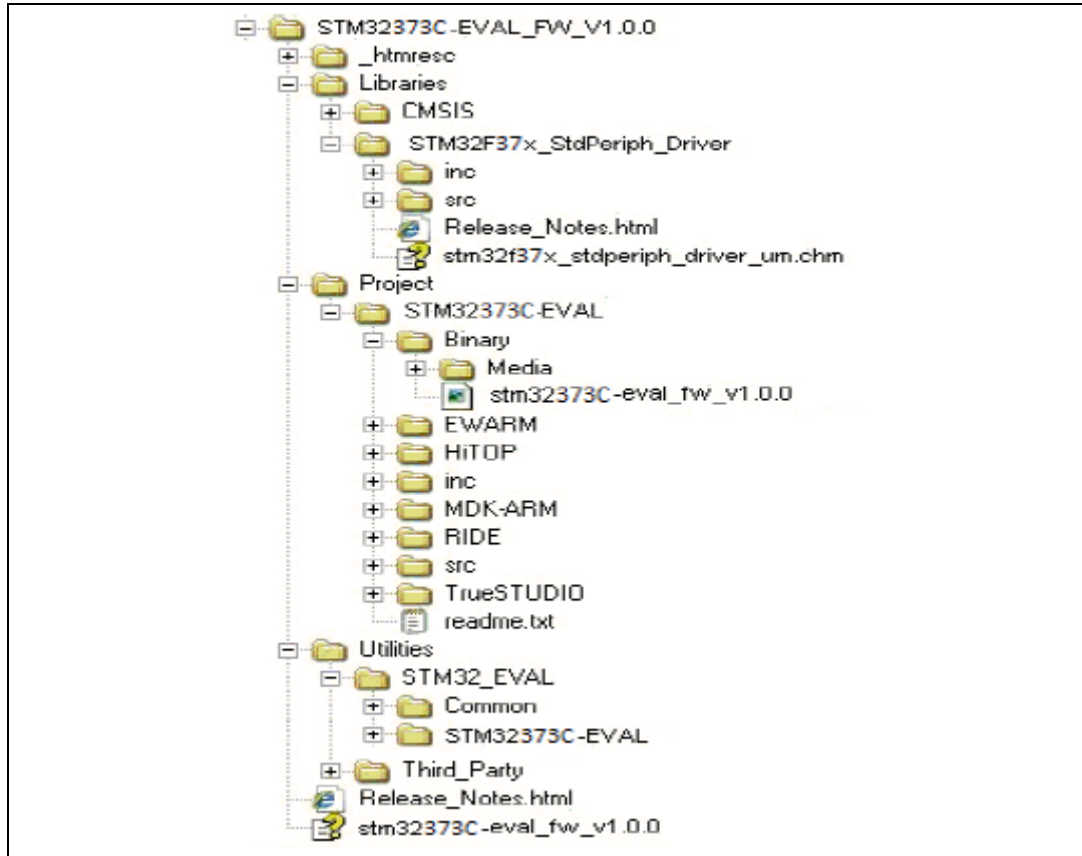


Pressing KEY then displays a message showing the STM32373C-EVAL demo version on the LCD screen.

3 STM32373C-EVAL demonstration package

The STM32373C-EVAL demonstration is supplied in a single zip file. The extraction of the zip file generates a single folder, STM32373C-EVAL_FW_VX.Y.Z, which contains the subfolders shown in [Figure 92](#) and described below.

Figure 92. STM32373C-EVAL demo package directory tree



3. **Libraries:** contains all the subdirectories and files that make up the core of the STM32F37x Standard Peripheral library V1.0.0:

CMSIS

- CM4\CoreSupport: contains the Cortex-M4 files
- CM4\DeviceSupport\ST\STM32F37x: contains the STM32F37x CMSIS layer files

STM32F37x_Stdperiph_Driver

- *inc* subfolder contains the Standard Peripheral library header files
- *src* subfolder contains the Standard Peripheral library source files

4. Project**STM32373C-EVAL**

- Binary: contains the binary image of the demonstration that can be used to program the binary image to the internal Flash memory using IAP, plus the media files required to run the demo (Binary\Media)
- EWARM: contains preconfigured projects for the EWARM toolchain
- MDK-ARM: contains preconfigured projects for the MDK-ARM toolchain
- RIDE: contains preconfigured projects for the RIDE toolchain
- TrueSTUDIO: contains preconfigured projects for the Atollic toolchain
- TASKING: contains preconfigured projects for the TASKING toolchain
- inc: contains the Demo header files
- src: contains the Demo source files

5. Utilities

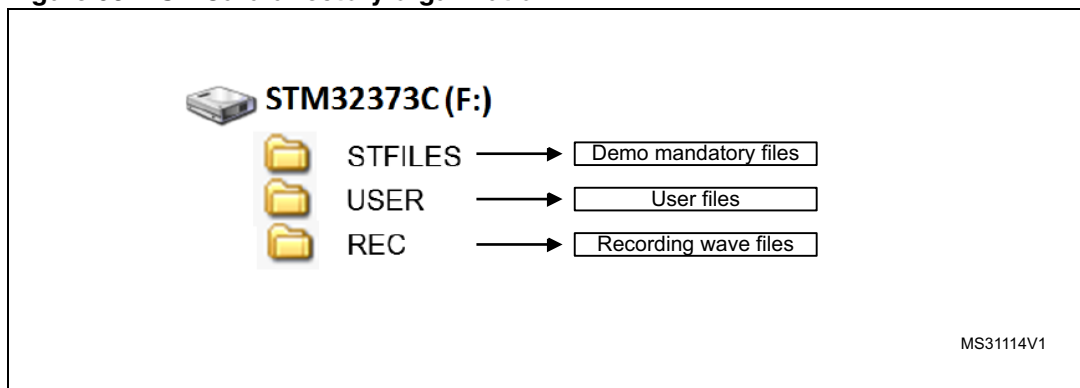
- STM32373C-EVAL: contains the LCD, and other STM32373C-EVAL board-related drivers

4 STM32373C-EVAL demonstration programming

4.1 Programming the media files

The STM32373C-EVAL board comes with a MicroSD card memory pre-programmed with the Audio and Image resources used by the demonstration. However, you can load your own image (*.bmp) and audio (*.wav) files in the USER directory, providing that these file formats are supported by the demonstration. For more details, please refer to [Section 2.4.2: Image Viewer submenu](#) and [Section 2.4.3: Audio](#).

Figure 93. SD Card directory organization



The default content of the media files (STFILES and USER directories) can be retrieved from the Binary\Media folder. So, if you want to reprogram the MicroSD card, you can copy the content of the Binary\Media folder to your own SD memory.

4.2 Programming the demo

You can program the demonstration using three methods.

4.2.1 Using Bootloader

To program the demonstration's binary image in the internal Flash memory, you have to use the `stm32373C-EVAL_FW_1.0.0.bin` file located under `Project\STM32373C-EVAL\Binary` with embedded Bootloader. For more details, please refer to Bootloader application note AN2606 *STM32™ microcontroller system memory boot mode*.

4.2.2 Using IAP

To program the binary image of the demonstration firmware in the internal Flash memory of the MCU, you have to use the `stm32373C-EVAL_FW_V1.0.0_offset_0x3000.bin` file located under `Project\STM32373C-EVAL\Binary` with IAP over USART. For more details, please refer to application note AN4045: *STM32F3xx in-application programming using the USART*.

4.2.3 Using preconfigured projects

1. Select the folder corresponding to your preferred toolchain (MDK-ARM, EWARM, RIDE, TASKING or TrueSTUDIO).
2. Open the STM32373C_EVAL project and rebuild all sources.
3. Load the project image through your debugger.
4. Restart the evaluation board (press B1: reset button).

5 Revision history

Table 6. Document revision history

Date	Revision	Changes
26-Sep-2012	1	Initial release.

Please Read Carefully:

Information in this document is provided solely in connection with ST products. STMicroelectronics NV and its subsidiaries ("ST") reserve the right to make changes, corrections, modifications or improvements, to this document, and the products and services described herein at any time, without notice.

All ST products are sold pursuant to ST's terms and conditions of sale.

Purchasers are solely responsible for the choice, selection and use of the ST products and services described herein, and ST assumes no liability whatsoever relating to the choice, selection or use of the ST products and services described herein.

No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted under this document. If any part of this document refers to any third party products or services it shall not be deemed a license grant by ST for the use of such third party products or services, or any intellectual property contained therein or considered as a warranty covering the use in any manner whatsoever of such third party products or services or any intellectual property contained therein.

UNLESS OTHERWISE SET FORTH IN ST'S TERMS AND CONDITIONS OF SALE ST DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY WITH RESPECT TO THE USE AND/OR SALE OF ST PRODUCTS INCLUDING WITHOUT LIMITATION IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE (AND THEIR EQUIVALENTS UNDER THE LAWS OF ANY JURISDICTION), OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

UNLESS EXPRESSLY APPROVED IN WRITING BY TWO AUTHORIZED ST REPRESENTATIVES, ST PRODUCTS ARE NOT RECOMMENDED, AUTHORIZED OR WARRANTED FOR USE IN MILITARY, AIR CRAFT, SPACE, LIFE SAVING, OR LIFE SUSTAINING APPLICATIONS, NOR IN PRODUCTS OR SYSTEMS WHERE FAILURE OR MALFUNCTION MAY RESULT IN PERSONAL INJURY, DEATH, OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE. ST PRODUCTS WHICH ARE NOT SPECIFIED AS "AUTOMOTIVE GRADE" MAY ONLY BE USED IN AUTOMOTIVE APPLICATIONS AT USER'S OWN RISK.

Resale of ST products with provisions different from the statements and/or technical features set forth in this document shall immediately void any warranty granted by ST for the ST product or service described herein and shall not create or extend in any manner whatsoever, any liability of ST.

ST and the ST logo are trademarks or registered trademarks of ST in various countries.

Information in this document supersedes and replaces all information previously supplied.

The ST logo is a registered trademark of STMicroelectronics. All other names are the property of their respective owners.

© 2012 STMicroelectronics - All rights reserved

STMicroelectronics group of companies

Australia - Belgium - Brazil - Canada - China - Czech Republic - Finland - France - Germany - Hong Kong - India - Israel - Italy - Japan - Malaysia - Malta - Morocco - Philippines - Singapore - Spain - Sweden - Switzerland - United Kingdom - United States of America

www.st.com